Design of Lightweight Alternatives to Secure Border Gateway Protocol and Mitigate against Control and Data Plane Attacks

Junaid Israr

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the PhD degree in Electrical and Computer Engineering

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Border Gateway Protocol (BGP) is the backbone of routing infrastructure in the Internet. In its current form, it is an insecure protocol with potential for propagation of bogus routing information. There have been several high-profiles Internet outages linked to BGP in recent times. Several BGP security proposals have been presented in the literature; however, none has been adopted so far and, as a result, securing BGP remains an unsolved problem to this day.

Among existing BGP security proposals, Secure BGP (S-BGP) is considered most comprehensive. However, it presents significant challenges in terms of number of signature verifications and deployment considerations. For it to provide comprehensive security guarantees, it requires that all Autonomous Systems (ASes) in the Internet to adopt the scheme and participate in signature additions and verifications in BGP messages. Among others, these challenges have prevented S-BGP from being deployed today. In this thesis, we present two novel lightweight security protocols, called Credible BGP (C-BGP) and Hybrid Cryptosystem BGP (HC-BGP), which rely on security mechanisms in S-BGP but are designed to address signature verification overhead and deployment challenges associated with S-BGP. We develop original and detailed analytical and simulation models to study performance of our proposals and demonstrate that the proposed schemes promise significant savings in terms of computational overhead and security performance in presence of malicious ASes in the network.

We also study the impact of IP prefix hijacking on control plane as well as data plane. Specifically, we analyze the impact of bogus routing information on Inter-Domain Packet Filters and propose novel and simple extensions to existing BGP route selection algorithm to combat bogus routing information.

Acknowledgements

It is a pleasure and honor to reach this milestone and I would like to thank Allah for giving me the health and opportunity to get this far in life. I would also like to thank many people who helped me unconditionally throughout this journey of my Ph.D.

I am indebted to my supervisor, Dr. Hussein T. Mouftah, for his valuable guidance, support and ideas. I am very fortunate to have had the company of Mouhcine Guennoun. Throughout my Ph.D., he and Dr. Mouftah provided encouragement, sound advice, good company, and several important ideas. I am privileged to have worked with both of them.

I wish to thank my entire family for providing me a loving environment. I know that this was made possible due to prayers of my elders, especially my late grandparents. A very special thanks to my parents, Israr Akhtar and Qaiser Israr, to whom I owe my life. My achievements in life are because of their sacrifices, prayers and encouragement. I wish to thank my wife Shazima, from the bottom of my heart for her unconditional love, endless support, patience, and continuous encouragement. My Ph.D. would not have been possible without her support and sacrifices. I am very grateful to Allah for blessing me with my children, Duaa and Waaiz, who inspired me to achieve this goal. I also want to thank my parents-in-law Izhar Athar and Shaheen Athar for their continuous prayers and support for me.

I want to thank my brothers, Tauseef Israr, Moeed Israr and Toqeer Israr who have been fantastic companions throughout my life. I would also like to thank my dear friends Imran Ahmad, Naheed Imran, Muhammad Asif, Hassan Sheikh, Fizza Sheikh, Viqas Zafar, Osman Hashmi and Omar Imran who always encouraged me and cheered me on at every step of this journey. Thanks also to my colleagues Dr. Tarek Saad, Dr. Zafar Ali and Dr. Siva Sivabalan for their constant feedback.

I will always need to remember and thank my aunt, Anjum Shakeel, who helped lay the foundation for my academic life and Damian MacLellan and Donald Bailey for giving me self-belief in my first year at Carleton University.

Last, but not least, I would like to thank David Ward for introducing me to my topic of research and for his invaluable advice and encouragement throughout this effort.

Table of Contents

List of Tables

List of Figures

List of Acronyms

ACR	Availability-Centric Routing
AP	Availability Provider
AS	Autonomous System
BGP	Border Gateway Protocol
CA	Certification Authority
C-BGP	Credible Border Gateway Protocol
CIDR	Classless Inter-Domain Routing
CPU	Central Processing Unit
DDoS	Distributed Denial of Service
DNS	Domain Name System
DOS	Denial of Service
DSUA	Documented Special Use Address
DV	Deferred Validation
eBGP	External Border Gateway Protocol
EGP	Exterior Gateway Protocol
GTSM	Generalized TTL Security Mechanism
HC-BGP	Hybrid Cryptosystem Border Gateway Protocol
HMAC	Hash-based Message Authentication Code
IANA	Internet Assigned Numbers Authority
iBGP	Internal Border Gateway Protocol
IDPF	Inter-Domain Packet Filters
IDRP	Inter-Domain Routing Protocol
IKE	Internet Key Exchange

IP	Internet Protocol
IPsec	Internet Protocol Security
IRV	Interdomain Route Validation
ISO	International Organization for Standardization
ISP	Internet Service Provider
MAC	Message Authentication Code
MD5	Message-Digest 5
MED	Multi Exit Discriminator
MOAS	Multiple Origin Autonomous System
NCC	Network Coordination Centre
pgBGP	Pretty Good Border Gateway Protocol
PHAS	Prefix Hijack Alert System
psBGP	pretty secure Border Gateway Protocol
PAL	Prefix Assertion List
PKI	Public Key Infrastructure
PKIX	Public-Key Infrastructure (X.509)
RA	Route Attestation
RCS	Route Credibility Score
RFC	Request for Comment
RIB	Routing Information Base
RIPE	Réseaux IP Européens (RIPE, French for "European IP Networks")
RIS	Routing Information Service
RV	Real-Time Validation
S-BGP	Secure Border Gateway Protocol
SHA	Secure Hash Algorithm

SLA	Service Level Agreement
soBGP	secure origin Border Gateway Protocol
SPV	Secure Path Vector
TBGP	Trusted Border Gateway Protocol
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TTL	Time To Live
VQ	Validation Query
VR	Validation Response

List of Symbols

$G(V,E)$	graph having V as the set of nodes and E as the set of directed links
$S_{a,t}$	set of ASes, from which an attacker in AS a can forge addresses to attack t
$C_{s,t}$	set of ASes, from which attackers can attack t using addresses belonging to s
τ	ratio of ASes
F	sub-set of nodes where the new enhanced security scheme is deployed
μ	coverage ratio
Φ	victim fraction
Θ	attack fraction
Ψ	victim trace fraction
r	ratio of spoofed BGP routing updates
$\Phi_{\mu}(r)$	success of IDPF in limiting the number of victims of IP spoofing in the presence of a percentage r spoofed BGP routing updates in the network
$\Omega(\mu)$	average performance of IDPF in the presence of a variable rate μ of spoofed BGP UPDATE messages
$\theta_{\mu}(r)$	strength of IDPF in limiting the number of attackers in the presence of a percentage r of spoofed BGP routing updates
$\alpha(\mu)$	average strength of IDPF filters in protecting the network against attackers
$\delta_{\mu}(r)$	effectiveness of IDPF filters in determining the true origin of spoofed packets in the presence of a percentage r of spoofed BGP routing updates
$\beta(\mu)$	average effectiveness of IDPF filters in determining the true origin of spoofed packets

X	score (IP address prefix origination)
Y	score (AS-PATH validation)
N	number of ASes in the network
$avgL$	average number of links per AS
$avgU$	average number of received updates
ps	size of the AS-PATH
n	number of hops to origin O
α_n	set of ASes that are n hops away from origin
x	ratio of trusted ASes
O	origin of the BGP UPDATE
NT_AS_n	*non-trusted* AS
NT_SigV_n	number of verifications performed by *non-trusted*
T_AS_n	*trusted* AS
T_SigV_n	number of verifications performed by *trusted* AS
$SigV_n$	number of verifications performed by *trusted* AS
$NT\_prefixV_n$	number of IP prefix validations performed by a *non-trusted* AS
$T\_prefixV_n$	number of IP prefix validations performed by *trusted* AS
$prefixV_n$	number of IP prefix verifications performed by an AS
z	number levels that constitute the network
m	malicious AS
v	victim AS

c	ratio of ASes with corrupted RIB
D	damage performed by IP prefix hijacking
NP	non-polluted AS
P	polluted AS
p_{ij}	transition probability from state i to state j
A_SigV_n	average number of asymmetric signature verifications performed by AS at hop n
S_SigV_n	average number of signature verifications performed by the AS at hop n
ρ	ratio of asymmetric signature verifications time units to symmetric verifications time units

Chapter 1

Introduction

1.1 Background

The Internet is a collection of interconnected networks, known as Autonomous Systems (ASes), which consist of routers and end devices. Each autonomous system is owned and managed by a legal entity, e.g. AT&T, Rogers, Sprint, etc. Typically, end devices originate or receive IP packets and routers are responsible for forwarding the packets along to the destination. In order to pass the packets along a consistent and optimal path, each router needs to know relative location (address) of other routers, and its reachability information. This information is used to make switching decisions on a per destination basis. It is the responsibility of the routing protocols to propagate the relative location and reachability information in the network.

Intra-domain routing is used by the set of routing systems operating within each autonomous system, while inter-domain routing is used to propagate reachability information between multiple autonomous systems. Since late 1980s, Border Gateway Protocol (BGP) [REK95] has been the routing protocol for both intra-domain communication (internal Border Gateway Protocol or iBGP) and inter-domain routing information exchange (external Border Gateway

Protocol or eBGP). Since BGP is the de facto routing protocol in the Internet, stability and security of the Internet heavily relies on the stability and security of BGP.

BGP is a path vector protocol, whose participants are known as BGP speakers. BGP speakers exchange routing information between each other. This routing information consists of an IP address prefix and a set of attributes that provide additional routing information associated with the address prefix. BGP heavily relies on an attribute, known as the AS-PATH. The AS-PATH captures an ordered list of ASes that were traversed by the BGP message from the AS that originated this update up to the current AS. The number of elements in the path is the AS-PATH length. Where a BGP speaker is presented with multiple paths to the same address prefix from a number of peers, the BGP speaker selects the “best” path to use by minimizing a distance metric across all the possible paths. The distance metric used by BGP speakers is the AS-PATH length. This BGP-selected route object is used to populate the local forwarding table. The BGP speaker then assembles a new route object by taking the locally selected route object, attaching locally significant attributes and adding its own AS value to the route object’s AS-PATH vector. This route object is then announced to all BGP peers.

1.2 Motivation

Since BGP related routing outages in the Internet are not uncommon, the topic of addressing security concerns in BGP continues to evoke a lot of interest both in the research community as well as in the industry. Although several proposals have been put together to address security concerns, reasons such as heavy computational overhead, technical difficulties in piecemeal adoption, incomplete security, and costs associated have, in some way or another, hampered adoption of these techniques. As such, there remains room and appetite for fresh ideas for securing BGP (both modifications of existing ones or drastically different from existing ones). It is important to learn from existing solutions and determine what attributes are absolutely key in ensuring a smooth transition and adoption to a secure BGP environment and include those in any new proposal.

In terms of security, Secure BGP (S-BGP), [KEN00a], is widely considered as the most comprehensive solution. It incorporates several security procedures that make it very difficult for any malicious AS in the system to inject bogus BGP routing information. However, S-BGP has not been deployed. There are significant concerns with regards to computational load of validation of signatures, significant memory requirements with the addition of a growing attribute of a signature and a public key identifier for every AS in the AS Path. In some cases, there are considerations for multi-path BGP as well. In reality, some of these impacts have not been fully quantified in studies to date.

S-BGP requires significant upgrades to existing hardware and software throughout many ASes. Except for a few major ASes who may be able to afford such expensive upgrades in the interest of BGP security, there is not a very strong motivation for smaller AS operators to incur such significant capital costs. Also, the Internet is very sensitive to changes in BGP and operators have generally shied away from adopting S-BGP because of the nature of BGP changes involved.

Considering the above, we explored lightweight secure alternatives to S-BGP. In terms of security procedures, we did not intend to propose a dramatically different approach as compared to S-BGP. We opted to retain and leverage security procedures in S-BGP and sought enhancements to improve its chances of adoptability both in terms of deployment considerations as well as performance costs.

We note that, in broad terms, there are two types of ASes in the Internet; these are “transit” ASes and “stub” ASes. A transit AS mostly propagates BGP routes on behalf of other ASes, while a stub AS either originates or terminates BGP routes. It is important to note that approximately 15% of the entire Internet today consists of transit ASes, while the rest are mostly stub ASes [HUS11d]. The transit ASes are typically very large entities with strong financial clout and strict security requirements as well as Service Level Agreements (SLAs). The stub ASes are typically customers of the transit ASes and are much smaller in size and scope. It was our intention to design proposals which can leverage this significant mismatch, not only in terms of requirements of the two types of ASes, but also the technical capabilities and their relative population (15% versus 85%).

We believe that transit ASes would be willing to and capable of upgrading their networks despite the costs, and the stub ASes would participate and provide minimal services needed to make the system work.

1.3 Objectives

We set out to study existing infrastructures available to secure BGP, both at the control plane level as well as the data plane level. We want to demonstrate and understand the impact of security breach on data plane protection techniques, such as Interdomain Packet Filters (IDPF). Once the impact is established, it is our objective to design multiple lightweight alternative protocols to secure BGP and address the vulnerabilities discussed above.

We seek to propose lightweight secure alternatives to existing BGP security proposals like S-BGP. At a very high-level, our design objectives are to leverage basic security mechanisms in S-BGP but propose significant changes in the operational model to make it more attractive to deploy. We have recognized the fact that 85% of the entire Internet today consists of stub ASes which are smaller entities, not readily capable to upgrading to S-BGP. The goal is to minimize the impact on such ASes and implement heavier security procedures on the transit ASes who are not only more strategic to the operation of the Internet but also more willing and capable of costly software and hardware upgrades.

In terms of the designs themselves, it is our objective to keep them as lightweight as possible as compared to existing security mechanisms available, while at the same time fulfill security gaps in BGP. We also want to demonstrate the effectiveness of the proposed architectures, both in terms of computational costs as well as security considerations. It is our goal to develop original analytical models for the new architectures in order to quantify the benefits of the proposed schemes. In order to confirm the validity of these analytical models, we also need to design a new simulator and obtain results to be able to compare with the analytical models.

It is also our objective to provide a balanced analysis in terms of benefits as well as potential areas of concerns for the proposed schemes. To this end, we would develop novel analytical and simulation models to study the security aspects of the proposal.

1.4 Contributions

The main research contributions of this thesis are as follows:

1. A novel, lightweight and computationally inexpensive extension to existing BGP route selection algorithm to incorporate security procedures when receiving and selecting BGP UPDATE messages, has been introduced.
2. A novel lightweight security protocol, called C-BGP, which introduces the concept of *trusted* ASes in BGP, has been provided.
3. A novel, hybrid cryptosystem and lightweight security proposal, called HC-BGP, employing both symmetric and asymmetric cryptography, has been presented.
4. Original analytical and simulation models have been developed for C-BGP that can determine the following parameters of a network of N ASes:
    - Average number of signature verifications
    - Average number of IP prefix verifications
    - Average number of public keys required
5. Original analytical and simulation models have been developed for HC-BGP to determine the following parameters in a network of size N ASes:
    a. Average number of asymmetric signature verifications
    b. Average number of symmetric signature verifications
    c. Average number of validation queries performed by *non-trusted* ASes in real-time mode.
6. An analytical as well as simulation framework has been provided to study the security aspects of the C-BGP. We developed new metrics to capture the security impact of misconfigured or malicious *trusted* AS(es) in the network.
7. A detailed simulation analysis to ascertain the impact of spoofed BGP UPDATE messages on BGP Inter-Domain Packet Filtering technique has been presented.

8. A simulation analysis has been carried out to study the impact of new BGP selection algorithm in improving the effectiveness of Inter-Domain Packet Filtering techniques in the presence of spoofed BGP UPDATE messages
9. The processing speed aspects of HC-BGP have also been evaluated using simulations.

1.5 Thesis Outline

The remainder of this thesis is organized as follows. In Chapter 2, we present a state of the art survey on BGP and mainly its security aspects and techniques. Since securing BGP has been a very active research area, we focus on major contributions made in the past several years and also briefly discuss their pros and cons.

In Chapter 3, we discuss Inter-Domain Packet Filtering proposal and present a simulation analysis on the negative impact of spoofed BGP UPDATE messages on this proposal. We also present a novel extension to existing BGP route selection algorithm to adapt to security requirements. We apply this proposal and use the simulations to demonstrate the improved effectiveness of Inter-Domain Packet Filtering techniques in the presence of spoofed BGP UPDATE messages.

In Chapter 4, we present Credible-BGP (C-BGP) which enhances S-BGP by adding a trust factor in the network. We also present an original mathematical model as well as a simulation model to determine multiple performance metrics for the proposal.

In Chapter 5, we study the security effectiveness of the C-BGP. We present a novel analytical and simulation framework to analyze the security aspects of the proposed schemes as compared to existing protocols. We present new metrics to capture the security impact of misconfigured or malicious *trusted* AS(es) in the network.

In Chapter 6, we propose Hybrid Cryptosystem BGP (HC-BGP), which employs symmetric and asymmetric cryptography systems to secure BGP UPDATE messages. We also relieve *non-trusted* ASes of signature verification operations. We present a novel mathematical model and a simulation model to determine multiple performance metrics for this enhanced proposal.

In Chapter 7, we discuss our contributions, results, recommendations and ideas for future research.

1.6 List of Publications

1. J. Israr, M. Guennoun, and H. T. Mouftah. "Mitigating IP Spoofing by Validating BGP Routes Updates". *International Journal of Computer Science and Network Security*, Vol. 9, No. 5, pp. 71-76, May 2009.
2. J. Israr, M. Guennoun, and H. T. Mouftah. "Credible BGP- Extensions to BGP for Secure Networking". *In the Proceedings of the Fourth IEEE International Conference on Systems and Networks Communications (ICSNC 2009)*, pp. 212-216, September 2009.
3. J. Israr, M. Guennoun, and H. T. Mouftah. "Credible-BGP: A Hybrid Cryptosystem to Secure BGP". *In the Proceedings of the IEEE Global Communications Conference (GLOBECOM 2010)*, CIS02.2.1 - CIS02.2.6, December 2010.
4. J. Israr, M. Guennoun, and H. T. Mouftah. "Analysis of Impact of Trust on Secure Border Gateway Protocol". *In the Proceedings of the Sixteenth IEEE Symposium on Computers and Communications (ISCC'11),* pp. 1099-1104, June 2011.

5. J. Israr, M. Guennoun, and H. T. Mouftah. "Security Analysis of C-BGP: A light Alternative to S-BGP", submitted to *IEEE Globecom 2012 - Communication and Information System Security Symposium*, December 2012.
6. P. Mohapatra, J. Scudder, D. Ward, R. Bush, R. Austein. “BGP Prefix Origin Validation”. IETF Draft, July 2011, http://tools.ietf.org/id/draft-ietf-sidr-pfx-validate-01.txt., Contributors: R. Fernando, K. Patel, M. Kohno, S. Miyakawa, T. Mizuguchi, T. Yoshida, R. Housley, J. Israr, M. Guennoun, and H. T. Mouftah.

Chapter 2

Survey of Related Work

2.1 Introduction

BGP related routing outages in the Internet are not uncommon; see Table 2-1 for a history of routing related outages in the Internet. Hence, securing BGP remains a very active research topic in the research community as well as in the industry. Recently, an entire working group has been setup at Internet Engineering Task Force (IETF) to discuss and adopt standards to secure BGP [SID10]. Over the past several years, a number of BGP security proposals have been submitted and some have even been prototyped; see [BUT10],[HUS11a],[NIC10],[OLI09],[ORT09] and [KUH09] for detailed surveys of the proposals. However, none has been deployed in the Internet today.

Currently, majority of the ISPs (ASes) prefer to implement either local or small scale solutions to protect BGP from various attack types. These solutions typically include protection of the underlying TCP connection via MD5 signatures and defensive filtering of BGP announcements. While these solutions work for most basic forms of BGP attacks, they do not protect against more complex attacks targeting the BGP route decision making process itself. It is extremely important for BGP to determine which routes are valid for it to be able to distinguish normal mode of operation versus an attack mode.

In this chapter, we will be presenting background material and survey work for advances in BGP and its security. We shall first discuss design and operation of BGP. We will then discuss different techniques/aspects (cryptographic techniques, protection of BGP session, defensive filtering, and routing registries) to secure various aspects of BGP. Then, we shall present a detailed overview of the four most comprehensive approaches to BGP security: S-BGP, soBGP, IRV and BGPSEC. We shall also briefly discuss a number of other schemes that have been proposed in the last several years.

Table 2-1: History of routing outages in the Internet [MUR09].

Year	Incident
Apr. 1997	AS 7007 announced routes to all the Internet
Apr. 1998	AS 8584 mis-announced 100K routes
Dec. 1999	AT&T's server network announced by another ISP – misdirecting their traffic (made the Wall Street Journal)
May 2000	Sprint addresses announced by another ISP
Apr. 2001	AS 15412 mis-announced 5K routes
Dec. 24, 2004	Thousands of networks misdirected to Turkey
Feb. 10, 2005	Estonian ISP announced a part of Merit address space
Sep. 9, 2005	AT&T, XO and Bell South (12/8, 64/8, 65/8) misdirected to Bolivia [the next day, Germany – prompting AT&T to deaggregate]
Jan. 22, 2006	Many networks, including PANIX and Walrus Internet, misdirected to NY ISP (Con Edison (AS27506))
Feb. 26, 2006	Sprint and Verio briefly passed along TTNET (AS9121) announcements stating that it was the origin AS for 4/8, 8/8, and 12/8
Feb. 24, 2008	Pakistan Telecom announces /24 from YouTube
Mar. 2008	Kenyan ISP's /24 announced by AboveNet
Sep. 08, Nov. 08, Jan. 09	Full BGP table leaks, e.g., Sep08 (Moscow), Nov08 (Brazil), Jan09(Russia)
Aug. 27, 2010	RIPE NCC's Routing Information Service (RIS) experimented with optional attributes in the Border Gateway Protocol (BGP). The experiment caused a massive increase in routing instability. Up to 1.4% of the Internet was affected by instability around the time of the experiment. [NEU11]

2.2 Design and Operation of BGP

The current version, BGP- 4, was first deployed within the Internet in 1993. The RFC describing this protocol, RFC1771 [REK95], was published in March 1995, and subsequently refined with the publication of RFC4271 in January 2006 [REK96]. The protocol has been stable for some years now. Across the deployment lifetime of BGP-4, the Internet has grown from an average of 20,000 distinct routing entries in 1993 to some 300,000 routing entries in 2009 [HUS09a].

2.2.1 BGP Messages

The repertoire of defined messages are: an OPEN message to start a BGP session, an UPDATE message to exchange reachability information, a NOTIFICATION message, which is used to convey a reason code prior to termination of the BGP session, KEEPALIVE messages, used to confirm the continued availability of the BGP peer, and ROUTE-REFRESH request messages to request a resend of the routing information. BGP uses an explicit OPEN message to commence a BGP peering session. Once the session is active, BGP operates via the exchange of UPDATE messages. Each UPDATE message contains a set of address prefixes that are unreachable (withdrawals), followed by a set of common route object attributes, and a set of address prefixes that share this set of attributes (announcements). The withdrawn prefixes are those prefixes where the local BGP speaker sees no reachability, and now wants to withdraw a previous advertisement of reachability. No routing attributes are associated with these withdrawn prefixes. The announced prefixes are those prefixes where the local BGP instance has an updated view of the reachability of a prefix that was previously withdrawn or unannounced, or has an updated view of the routing attributes of the locally selected “best” route for a prefix.

2.2.2 AS-PATH Attribute

BGP binds together the concept of network address blocks and autonomous systems into a path vector-based routing technology. Every route object represented within a BGP-4 route database contains an address prefix and an associated path vector of AS values. BGP does not indicate the precise path a packet should follow within an AS, nor does it maintain a complete map of the topology of the Internet at a link-by-link level. BGP uses a level of abstraction which views the Internet as a set of per-AS routing domains, and the role of BGP is to maintain a routing map of the network at this AS level, associating every reachable address prefix with an AS transit path from the current location to the address prefix's originating AS.

One of the most important route object attributes in BGP is the AS-PATH attribute. As address prefix reachability information traverses the Internet in the form of individual route objects in BGP, this BGP routing information is augmented by the list of ASes that have been traversed thus far, forming the AS-PATH attribute. Each BGP speaker adds its own AS value to the route object's AS-PATH attribute when passing the route object through an eBGP session. This AS-PATH attribute allows straightforward suppression of the looping of routing information, using the simple algorithm that a local AS will reject any forwarded route object that already contains its own AS in the AS-PATH attribute. Also the length of the AS-PATH vector forms the BGP route metric. A local BGP system, when attempting to select one of a number of potential route objects that refer to the same address prefix, will, in the absence of any local policy directive, prefer the route object with the shortest AS-PATH length.

In addition to undertaking the role of path metric and loop detector, the AS-PATH attribute serves as a versatile mechanism for policy-based routing, where a local AS can alter the default preferences for route selection based on local policy settings coupled with pattern matching rules to be performed on the AS-PATH.

2.2.3 BGP Route Selection Process and Routing Policies

A BGP speaker may receive two or more announcements for the same address prefix from different peers. The "best" announcement is selected as the locally used announcement, and this announcement is the one that is announced to its BGP peers. BGP defines an ordered sequence of comparisons to determine which route object is selected by the local BGP speaker:

- Select the route object with the highest value for local-preference attribute value
- Select the route object shortest AS-PATH attribute length
- Select the lowest multi-exit discriminator (MED) attribute value
- Select the minimum IGP cost to the next hop address given in the route object
- Select eBGP over iBGP-learnt routes
- If iBGP select the lowest BGP identifier value.

Although a network administrator's usually employs routing policies depending on his needs [GRI05],[WAN03b], within the generic BGP route selection process the highest priority selection rule is that a route for a more specific address prefix is to be preferred over that of a covering prefix.

2.2.4 Vulnerabilities of BGP

BGP has several well-known vulnerabilities. These vulnerabilities are the direct consequences of three fundamental weaknesses in the BGP and the inter-domain routing environment [HUS10]. First, there is no mechanism to check the integrity, freshness and source authenticity of BGP messages. We note that mechanisms like the TCP MD5 “signature” provide these functions on a point-to-point basis. What we believe is missing is a way to provide these security services for the AS-PATH info that is passed, transitively, through BGP UPDATE messages. Second, BGP doesn’t offer any mechanism to verify the authenticity of an address prefix and an AS origination of this prefix in the routing system. Last, BGP doesn’t provide any way to guarantee that the attributes of a BGP UPDATE message have not been tampered with along the AS-PATH.

There are several different ways of exploiting these vulnerabilities. For example, a malicious user can easily pose as a trustable participant and inject false routing information in the network. This has the potential to wreak havoc on the Internet infrastructure as traffic can easily be redirected to unintended networks and cause major network outages. Moreover, with the current BGP infrastructure, such attacks may not be traceable to their origins which makes it very attractive proposition for the attackers.

2.3 Incentives to Attack BGP

When BGP was originally designed for the Internet, it was under the assumption that routers and end devices *inside* the network could be trusted and that any security threat resided *outside* the network. However, there are a number of factors which have challenged this assumption and resulted in multiple ASes knowingly or otherwise choosing to deviate from correct operation of BGP. These factors include economic gains, depletion of IPv4 address space, misconfigurations, eavesdropping, denial of service, stealing unallocated addresses and diverting traffic for malicious purposes, and source IP address spoofing, again for malicious purposes.

2.3.1 Economic Incentives

Each AS is typically owned by profit seeking legal entity. BGP requires each AS to exchange their connectivity information such that each AS can make best decision about the path to choose when forwarding packets from source to destination. For BGP to operate correctly, each AS must continue to cooperate with others, even though the ASes may be direct competitors of each other from economic point of view. For profitability reasons alone, an Internet Service Provider (ISP) might misrepresent network performance and thus attract more of traffic from its paying customers. It may also monitor traffic flow through its competitor AS and re-adjust its BGP policies according to its own economic interests.

2.3.2 Depletion of IPv4 Address Space

The depletion of the IPv4 allocation pool has been an ever growing concern for last two decades since the Internet started to experience dramatic growth. As IPv4 address space allocation runs out, it is becoming increasingly difficult and expensive to acquire new IPv4 addresses. Also, there is increased scrutiny for efficient usage of already allocated IPv4 address blocks. Some of the larger allocations which happened long time ago are not being efficiently used and there is evidence that several IPv4 addresses remain unused even though the address block containing those IPv4 addresses has been allocated/reserved to an AS several years ago [HUS11d]. The ongoing rapid growth of the Internet, combined with difficulty/cost associated with acquiring new IPv4 addresses and the fact that several allocated IPv4 addresses remain unused is a perfect recipe for new and smaller ASes operating in remote regions to illegally re-use IPv4 addresses that have been allocated to legitimate owner ASes.

2.3.3 Misconfigurations

It is very difficult, if not impossible, to eliminate human error. Most misconfigurations are caused by human error. A misconfiguration of the router includes, but is not limited to, incorrect addresses advertisements, improper route filtering, incorrect policy configurations etc. One of the more recent incidents happened few years ago in which Pakistan's state-owned telecommunications company cut YouTube off the Internet due to a misconfiguration [BRO08]. The misconfiguration allowed Pakistan Telecom to announce itself (via BGP) to as a legitimate destination for anyone trying to reach YouTube's range of Internet addresses. Even though the issue was detected within five minutes of it happening, it took more than two hours and intervention at multiple sites before access to YouTube was fully restored.

2.3.4 Eavesdropping, Denial of Service

This category falls under the umbrella of malicious ASes in the systems who are acting solely in self-interest. It can be used for security, espionage or other harmful purposes. Eavesdropping is achieved by influencing packet forwarding path such that traffic can pass through an eavesdropping location prior to final destination. In the context of BGP, Denial of Service (DoS) is achieved by overcoming routers in a particular AS or multiple ASes such that traffic to an address prefix is discarded at this AS. There are several flavours of BGP DOS attacks, most of which are discussed in [NOR04].

By stealing unallocated addresses and advertising them via BGP, there is an incorrect assertion of existence of addresses and forwarding paths to those addresses that should not exist in the network in the first place. Since there is no allocation registration information associated with these addresses, it allows malicious ASes to mount anonymous attacks rather easily.

2.3.5 Source IP Address Spoofing (Data-plane)

The lack of source IP address validation across multiple autonomous systems (AS) in the Internet makes it difficult to detect and prevent attackers from launching Distributed Denial of Service (DDoS) attacks using spoofed source addresses. Several popular Internet sites [RIC00] and Internet infrastructure [NAR02] have been attacked recently and such attacks have the potential to cripple the Internet. Detection and prevention of these attacks is often made more complicated by attackers employing source IP address spoofing. The idea is to forge the source IP address in the "attack" packets to that of another host in the system. This allows the attacker to pose as some other host and hide its actual identity and location, making it difficult to detect the actual attacker and to protect against it. As a result, attack detection techniques that rely on source address-based filtering become less effective when source address is spoofed by the attackers.

2.4 Cryptography and BGP Security

The following cryptographic techniques have often been used in several of the existing BGP proposals. We shall review these techniques to help understand their application in BGP security proposals.

2.4.1 Pair-wise Keying

This scheme intends to protect communication between a pair of nodes. It relies on the existence of a shared secret key which is agreed upon between the parties ahead of time. It is configured manually at each node in an offline basis and is used to establish authentication when communication is first established between the two nodes. This approach, while simple, provides significant scalability, complexity and management overhead to be practical in large scale deployments of BGP as the Internet.

2.4.2 Digest Algorithms

These algorithms (also called cryptographic hash functions) use an input text to generate a fixed length hash value. Currently, the most commonly used hash functions are Message-Digest algorithm 5 (MD5) [RIV92] and the Secure Hash Algorithm family, particularly SHA-1 [FIP02]. The cryptographic strength of the hash function is determined by the difficultly involved in converting the the hash back to original text value without any collisions (i.e., it is computationally infeasible to find two inputs with same output hash value).

2.4.3 Message Authentication Codes

A Message Authentication Code (MAC) is used to guarantee both the integrity (i.e. message was not tampered) of the message as well as authenticity of the sending node (i.e. sender had access to the secret key). The MAC is generated by feeding the secret key and the original message as an input into a mathematical function. The MAC is typically then appended to the original message and sent to the receiving node. The receiving node, with knowledge of the secret key, should be able to receive the message and generate its own local copy of the MAC. If the received MAC does not match the local copy of the MAC, then there is something suspicious about the message or its sender. HMAC [KRA97] variant is commonly used to generate a MAC.

2.4.4 Public Key Infrastructure

In a large scale routing system such as the Internet (with over 35,000 ASes [HUS11c]), the techniques above do not scale as they rely on a shared key between two parties. When more than 35,000 ASes need to exchange BGP messages in the Internet, management of pair-wise keys becomes a very significant challenge. Public key cryptography is one solution for key management on a global scale. In the context of BGP, every AS has a public key and a private key. The public key is distributed throughout the Internet and is known to all other ASes. The private key is kept secret. The assignment, delegation and distribution of public keys is handled by Public Key Infrastructure (PKI). Key distribution follows a hierarchical model. IANA is at the top of the hierarchy, followed by regional registries who assign keys to local ASes. There is significant ongoing research in this field to develop and make this infrastructure available for use in BGP [KEN00a][BUT10].

2.4.5 Public Key Cryptography

Public-key, or Asymmetric cryptography is heavily used in several security solutions to ensure message confidentiality and message integrity (message has not been modified in transit). To ensure message confidentiality, encryption is deployed by generating a ciphertext using the public key of the message recipient. In the context of BGP, when an AS receives this encrypted message, it can only decrypt if it has the associated private key.

Digital signatures are used to guarantee message integrity. A hash is generated with the private key of the AS sending the message and sent along with the message. The recipient AS must query the PKI to retrieve the public key of the sending AS and then use that public key to first decode the message and then compare the locally generated hash with the received hash value.

2.4.6 Certificates and Attestations

Certificates and attestations are essential parts of PKI and almost always form the basis of most of the proposed BGP security solutions. Attestations are used to prove that an AS owns a certain address prefix and is authorized to advertise it in the Internet. Attestations carry information on the ownership of address blocks and the ownership/delegation hierarchy. IANA is the ultimate root for all address allocation. It can delegate a large address block to an organization who can further delegate the address blocks to smaller organizations. The attestation can include this delegation hierarchy and will be digitally signed by the attesting AS or organization to ensure attestation integrity. The delegation hierarchy chain can be verified at each link, back to the source of the original delegation.

In order to verify the attestation, the public key of an AS is required. This key is accessed through a PKI using digital certificates. Digital certificates are issued by issued by a Certification Authority (CA) and contain both the public key of the AS and a signature confirming the validity of the certificate. The CA also follows hierarchical model with IANA at the top issuing a root certificate. The second/third level certificates are signed by an ISP or a national or regional registry that issues an AS number to the organization.

2.5 Protection of BGP Session

A comprehensive survey of techniques available to secure BGP session is documented in [BUT10]. We will briefly review some of the techniques mentioned in this survey. In order to protect BGP session between two ASes, you need to protect both the underlying TCP session as well the as the BGP session. Here are some techniques available to provide this multiple layer of protection.

2.5.1 MD5 Integrity

Use of MD5 digest [RIV92] based MAC as a TCP extension has recently been proposed to enhance BGP security. The protection is achieved by including an MD5 keyed digest [KRA97] of the TCP header and BGP data in each BGP message. The digest is generated only by someone with access to the secret key and this guarantees the authenticity of the packet data. There have been alterations proposed to hash all or part of the TCP and BGP data message using one or more keys [HEF98]. This is to address some of the concerns associated with spoofing and hijacking inherent to TCP [BEL89], [GRE02].

2.5.2 Session and Message Protection

There have been proposals submitted to protect BGP session and its messages. The most notable contribution in this area is by Smith and Garcia-Luna-Aceves [SMI96], [SMI98]. They proposed protection of BGP messages by encrypting all BGP data between peers (using a secret key shared by the peers) and adding sequence numbers to enforce a total ordering on the messages. They also propose to secure BGP UPDATE messages by adding an UPDATE sequence number or timestamp, a new path attribute, PREDECESSOR, that identifies the last AS before the destination AS, and digital signatures (signed by the peer) of all fixed value fields in the UPDATE messages. Although this proposal may provide sufficient security to BGP, it has significant management and adoption challenges. Management of pair-wise keys becomes very complex as the number of peers scales. Also, since this scheme proposes to change BGP implementations, this is a significant prohibitive barrier to adoption.

2.5.3 Generalized TTL Security Mechanism

The Generalized TTL Security Mechanism (GTSM) proposes to protect BGP nodes from remote attacks [35]. The underlying assumption in this approach is that most of the BGP peering nodes are adjacent to each other, i.e. they are one IP hop away. The proposal makes use of the time-to-live, or TTL, field in the IP header to determine whether the incoming IP packet traversed more than one hop from its source to its destination. The proposal calls for sending node to set the TTL of an IP packet to its maximum value of 255. When a BGP peer receives a packet, it checks the TTL and if this value is less than 254 (decremented by one), the packet is flagged or dropped.

The effectiveness of the proposal is limited, especially against complicated attacks as it does not defend against subverted peers sending malicious information.

2.5.4 IPsec

IPsec is a suite of protocols that provide security at the network layer [KEN05], [THA98] by defining methods for encrypting and authenticating IP headers and payload, and providing key management services. The Internet Key Exchange (IKE) protocol deals with the issues of dynamic negotiation of session keys [KAU05]. In the context of BGP, it provides the security guarantees for BGP sessions, e.g., authenticity of data, integrity, message replay prevention, and data confidentiality. IPsec sessions implement the required security between peers. However, it is not sufficient to protect against widespread attacks. IPsec is garnering much attention in recent times and is being adopted in multiple security applications to secure peer communications. It also forms the basis for the more significant BGP security solutions.

2.6 Defensive Filtering

The concept behind defensive filtering is to filter out unacceptable BGP announcements received from a peer. The BGP prefix announcements may be deemed as unacceptable due to the fact that the IP addresses being announced are not allowed to be advertised in the network. The following classes of IP addresses are good candidates for defensive filtering:

- Documented Special Use Address (DSUA) prefixes (e.g., loopback addresses).
- Bogons or martians (advertisements of address blocks and AS numbers with no matching allocation data). There are lists maintained (e.g. CIDR report [BAT08] which can be consulted for construction of route filters for bogons.

- Private AS numbers [STE08] or unrealistically long AS-PATHs.
- Very small subnets (i.e., smaller than a /24 block of 256 addresses). These are filtered to limit the size of the global routing tables.

Some ASes may choose to limit the total number of accepted prefixes from a peer. This is to protect against arbitrarily large growth of BGP routing table which may lead to memory exhaustion and also to prevent a neighboring AS from advertising disaggregated routes.

Defensive filtering can also be used in an intelligent way to perform filtering of BGP UPDATE messages which may be syntactically valid but are announcing BGP routes which are not plausible given the topology constraints. For example, defensive filtering can be intelligently applied by ASes for BGP UPDATE messages received by their customers, especially stub AS customers that do not provide transit service for others. In normal circumstances, a BGP update message from stub AS customer should only contain customer route information in the AS-PATH. If the AS-PATH contains the AS number of another ISP, then this update message is a candidate for defensive filtering with the presumption that the customer stub AS incorrectly propagated a provider learnt route. Similarly, ASes can filter BGP UPDATE messages from customers for prefixes which are not owned by the customers.

Although defensive filtering remains a very powerful way to reduce the size of BGP table and maintain BGP security, it is a manual process, with significant management overhead. Although it is widely deployed, the filters are based on heuristics and are not necessarily always up-to-date and there is no easy way to determine where it is not being used. Caesar and Rexford [CAE05] and Nordstrom and Dovrolis [NOR04] have investigated the effects of BGP routing and filtering policies.

Also, defensive filtering fails in scenarios, when a malicious AS is not directly connected but is several hops away. It is quite possible to be a target of an attack from an AS connected to ISPs that does not filter routes, or by a malicious user who has compromised a router in the ISP's network.

2.7 Routing Registries

The concept of routing registries is to produce a shared global view of "correct" routing information. Each AS would be required to populate a secure registry service with details of their policy and topology information. This registry can be queried by other ASes and it would make it easier to detect and reject invalid routes being advertised by an immediate peer or even by an attacker multiple ASes away. It may also be used to construct better defensive filters. The main obstacle in adoption of registries is the reluctance of ASes to publish their private policies and topology information into a public registry. Also, there is a need to make sure that the registry itself is secure and accurate. There have been proposals made in this field, most notably by Blunk et al. [BLU05].

2.8 BGP Security Architectures (S-BGP, soBGP, IRV, BGPSEC)

In this section, we discuss four comprehensive approaches to BGP security in terms of the increasing flexibility afforded to the user: S-BGP, soBGP, IRV and BGPSEC.

2.8.1 Secure BGP (S-BGP)

Among the myriad of techniques proposed to provide security in BGP, Secure BGP (S-BGP) has been the most complete contribution to date. The main goal of S-BGP is to protect the AS-PATH from modification and truncation, and to prevent unauthorized advertisements of an IP prefix. It makes use of public key certificates to communicate authentication data. Theses certificates are used to bind cryptographic information to an identity such as an organization. Any entity that is in possession of the public key certificate can validate information digitally signed with the private key associated with the public key [RIV78]. Important elements of S-BGP, notably the notions of the PKI, have been adopted by the SIDR working group and regional registries [SID10].

S-BGP ensures that all protocol update messages are valid using public key certificates. The validation consists of checking the integrity and freshness of data passed between ASes, and that the BGP speaker from which the update is originated is authorized to make the announcement on behalf of the Autonomous System. Early work in S-BGP called for a pair of PKIs used to delegate address space and AS numbers, and to associate particular network elements with their parent ASes, while later work collapsed this to one hierarchy [SEO01]. IPSec is used in S-BGP to secure the communication between BGP speakers [LYN03].

S-BGP supports two types of digitally signed statements called attestations. Address attestation is a signed statement that delegates the right to originate an address prefix from the owner (AS, organization) to another AS. The right to make an address prefix announcement can be checked by issuing an out of band query to the PKI repository. A route attestation is a signed statement that is carried by a BGP update. Each AS that receives the BGP update will sign the route and the previously signed attached signatures. The route attestation enables a BGP speaker to verify that the AS-PATH attribute is indeed the true sequence of ASes that the BGP update has traversed. The BGP speaker can validate not only the path, but also that a) the ASes were traversed in the order indicated by the path, and, b) no intermediate ASes were added or removed by an adversary. Figure 2-1 shows a simplified use of route attestations as they propagate between routers.

S-BGP provides a strong security to the BGP. However, there are a number of issues associated with it. First, because of the amount of data and number of signers, validation can be costly in terms of computation, storage and protocol data exchange [NIC02]. Simulation assessments [NIC04] of S-BGP show that path convergence times could as much as double with S-BGP. It should be stated that this overhead can be reduced by optimizing the protocol to only validate messages when they are selected as preferred BGP paths.

Second, the substantial storage requirements for route attestations have also been noted [KEN00b].

Also, assumptions relating to the environment, in which S-BGP must operate, severely impact the convergence of the Internet. The adoption of the protocol requires an upgrade to most of the Internet routers to enable them to support the computationally expensive protocol operations.

Last, the protocol is not flexible enough to deal with missing or out of date information in the PKI. As a matter of fact, it’s hard to guarantee the completeness and accuracy of the PKI repository in the presence of more than 35,000 ASes in the Internet and millions of address prefixes [UORV]. These protocol limitations have urged the research community to seek alternative solutions to secure BGP.

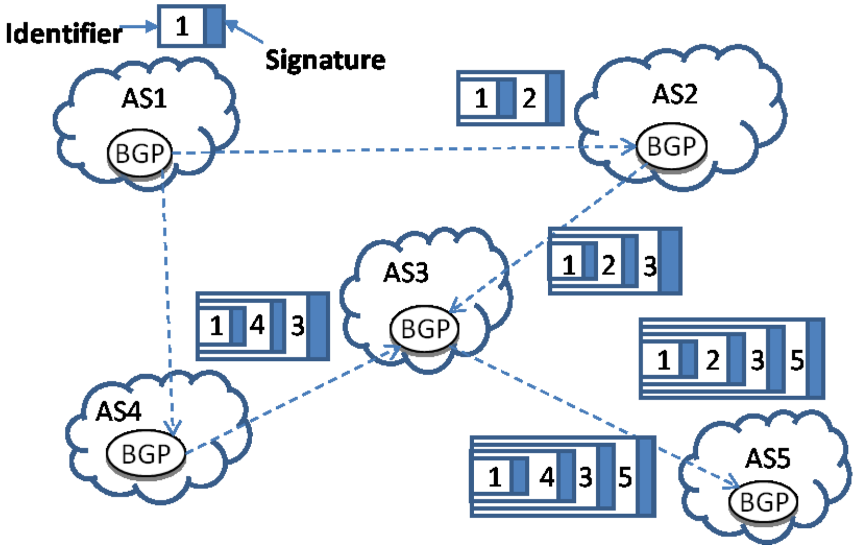

Figure 2-1: “onion-style” route attestation in S-BGP [BUT05].

2.8.2 Secure Origin BGP (soBGP)

Secure Origin BGP (soBGP) [NG04] is an attempt to strike a pragmatic balance between the security processing overhead and the capabilities of deployed routing systems and security infrastructure. soBGP uses similar concepts as S-BGP for address prefix validation. The difference is that S-BGP uses a hierarchical PKI, whereas, soBGP uses the concept of a web of trust to validate the binding certificates. soBGP defines a PKI for authenticating and authorizing entities and organizations. The PKI manages three types of certificates as follows:

1. A certificate to bind a public key to each soBGP-speaking router.
2. A certificate to provide details on policy, local network topology and protocol parameters. The local topology information is used by the router receiving the certificate to construct a topology database reflective of the router’s view of the network which is then used to validate received routes. Any UPDATE with a path that violates the AS topology is dropped. Each AS signs and distributes its local topology (i.e., its peers) through the topology certificate.

3. A certificate to cover address ownership or delegation (similar to S-BGP).

soBGP exchanges information relating to security in a new BGP message called SECURITY message. This in-band mechanism to distribute origin authentication information is in contrast to the out-of-band method used by S-BGP.

S-BGP and soBGP are significantly different in terms of their approach for path authentication. In S-BGP, route attestations are sent with every BGP UPDATE message and the receiving node has a real-time view of the path taken by the message. In soBGP, the topology graph and corresponding database used by soBGP is essentially static, as the topology only changes when a new policy certificate is issued. There is a timing window where new topology may not be updated when an UPDATE is received with an AS-PATH that this is different from the one in the peer's topology database. There is a need for additional infrastructure to ensure that the topology updates are synchronized across all ASes.

Also, with soBGP, there is a risk of accepting BGP UPDATE messages with forged AS-PATHs. soBGP topology will allow receiving node to drop any received route which is not consistent with the routing topology. However, there is no way to guarantee that the BGP UPDATE message followed the route received in the BGP UPDATE message. This is a security hole since a malicious AS-PATH may still be accepted [BEL03a].

soBGP aims to reduce the computational overhead of validating signatures by authenticating long-term structural routing elements (such as organization relationships, address ownership, and topology) prior to establishment of BGP session. Data pertaining to these elements is signed, validated, and stored at the routers ahead of time to reduce run-time costs.

soBGP provides several deployment options which allow for tradeoffs between security and convergence. One option allows the operator to choose whether to verify routes before accepting them into the routing table or to accept routes and then verify their authenticity. Other options provide flexibility for validation of route itself. As compared to S-BGP, soBGP stands out in terms of better choice from ease of deployment point of view, but the number of options could introduce interoperability challenges [KEN03a]. Also, certificates used in soBGP are non-standard compared to the IETF PKIX certificates used in S-BGP.

2.8.3 Interdomain Route Validation

Interdomain Route Validation (IRV) is a decentralized comprehensive to secure BGP [GOO03]. In this model, there is an IRV server in each AS which can be queried by the BGP speaker to determine whether the received UPDATE message contains correct information. Each of the IRV servers contains information about local AS topology and prefix ownership. When information is needed for a particular remote AS, the local IRV server can directly contact the IRV server in this AS. In this model, there is no in-band validation of BGP messages, which removes any computational (impacting convergence) or storage costs. There is flexibility afforded to each AS to have its own algorithm for when to check the validity of UPDATE message. Also, IRV server can choose to query all or a subset of ASes along the AS-PATH in the UPDATE message, based on previous associations (e.g., ASes known to provide *trusted* information may not be queried). There is also an option to cache past queries to assist with debugging and failure detection. These options allow the BGP speaker flexibility to tradeoff between convergence and security, if they so choose. Local policy configurations on the BGP speaker dictate what data can be *trusted* or ignored and what must be validated via an IRV query.

The IRV server model is slightly better than routing registries because the AS retains control over the data, and is more likely to keep it up to date. However, the underlying assumption is that AS makes accurate assertions and the IRV server is never misconfigured. In order to guarantee authenticity, integrity, and confidentiality of queries and results, communication between IRV servers can be secured at using IPSec or TLS [DIE06].

The central limitation of IRV is that a network must be available for BGP speakers to communicate with the IRV server or for IRV servers to communicate with each other. This connectivity may only exist through the very route which needs to be validated (chicken and egg problem). This presents a problem in bootstrapping the protocol and in recovering from failure scenarios. Some of the possible solutions to this include using static routing to connect the IRV servers, or use gossip-style protocols [BAK72], or use optimistic routing (e.g., using received routes immediately and validating where possible).

2.8.4 BGPSEC

BGPSEC [LEP11a] is designed primarily to use digital signatures to protect the AS-Path attribute of BGP UPDATE messages. The signatures allow the BGP node to assess the validity of the AS-Path in UPDATE messages that it receives.

BGPSEC builds on top of Resource Public Key Infrastructure (RPKI) [LEP11b], which introduces the notion of validation of BGP messages. RPKI resource certificates are issued to ASes and the IP addresses (prefixes) owned by them, thereby, creating an association between ASes, IP prefixes and cryptographic keys that can be used to verify digital signatures. RPKI also specifies a digitally signed object, a Route Origination Authorization (ROA), which allows owners of IP addresses to authorize specific ASes to originate BGP routes for these IP addresses.

ROAs are intended to be used by BGP nodes to determine whether a received route was originated by an AS authorized to originate that route [HUS10] and [BUS11]).

ROAs allow ASes to protect themselves from certain mis-origination attacks. This can be achieved by implementing a local policy which prefers local routes with origins validated using RPKI data versus routes which cannot be validated. However, use of ROAs and RPKI does not protect against an attacker who could, for example, conduct a route hijacking attack by appending an authorized origin AS to an otherwise illegitimate AS Path. BGPSEC adds a new BGPSEC router certificate, which is used to bind an AS number to a public signature verification key, the corresponding private key of which is held by one or more BGP nodes within this AS. The BGP nodes within an AS can use private keys to sign on behalf of their AS. The certificates allow the receiving BGP node to verify that a BGPSEC signature was created by a BGP node located in a given AS.

BGPSEC also introduces a new optional attribute, called BGPSEC_Path_Signatures, which consists of a sequence of digital signatures, one for each AS in the AS Path of a BGPSEC UPDATE message. Each AS adds its own signature to the outgoing UPDATE message. The signature is so designed that the recipient can detect any tampering with the AS PATH or IP prefix information.

When originating a BGPSEC update message, BGP node creates a BGPSEC_Path_Signatures attribute containing a single signature. This signature protects, among other things, the IP prefix, the AS number of the originating AS and the AS number of the peer AS to whom the update message is being sent. When a receiving BGP node propagates the route to a peer, it adds a new signature to the BGPSEC_Path_Signatures attribute. The intention is to protect everything protected by the previous signature, plus the AS number of the new peer. Each BGP node also adds a Subject Key Identifier (SKI) to its BGPSEC Router certificate, which is used by a recipient to select the public key needed for validation.

For each signature received in BGPSEC update message, the BGP node first needs to determine if there is a valid RPKI Router certificate matching the SKI and containing the appropriate AS number. The BGP node then verifies the signature using the public key from this BGPSEC router certificate. If all the signatures can be verified in this fashion, the BGP speaker is assured that the update message it received actually came via the path specified in the AS-Path attribute. Finally, the BGP speaker can check whether there exists a valid ROA in the RPKI linking the origin AS to the address prefix being advertised. If such a valid ROA exists the BGP speaker is further assured that the AS at the beginning of the validated path was authorized to originate routes to the given prefix [LEP11a]. If all the signatures spanning the AS-Path in the BGPSEC message can be validated, then the receiver of the BGP message can confirm that that the route update traversed the inter-AS routing space in the same manner as is represented in the AS-Path attribute of the BGP UPDATE message.

With current BGP standards, an announced BGP route is considered valid until it is withdrawn or until the session with the peer AS responsible for announcing the route is terminated. This property of BGP makes it vulnerable to “ghost route” attacks, wherein a BGP node chooses not to propagate a route withdrawal in the hope of diverting misdirected traffic from its peering ASes. BGPSEC changes this behavior by having ASes set expiry times for route advertisements originated by them. This expiry time is the notAfter field in the End Entity (EE) certificate, which is used to sign the protocol update and the route is considered invalid once this time elapses. The originating AS must periodically re-advertise the route and reset the expiry timer of the associated digital signature.

BGPSEC relies on a model of incremental deployment in which direct peers of a BGPSEC speaking AS will be able to communicate with each other using BGPSEC. These adjacent ASes can offer to speak BGPSEC with their respective peer ASes, and so forth. This implies that BGPSEC speaking communities of ASes will be able to provide assurance on routes originated and propagated within the community of interconnected ASes. When a BGPSEC update message leaves a BGPSEC community, BGPSEC signature attributes of the route are stripped out so the peering ASes can treat the message like a normal BGP message. Similarly, periodic updates resulting from the expiry timer do not propagate beyond the BGPSEC community. Also, BGPSEC will not allow for packing of updates, where a number of IP address prefixes can be advertised via a single UPDATE message if they share a common collection of attributes, including the AS-Path. In BGPSEC, each update message would refer to a single prefix. This does have potential to increase BGP traffic in the Internet but it needs to be quantified.

There are several aspects of BGPSEC which warrant further study and quantification. These include amount of additional local storage needed on each node to store signatures, public key identifiers for ASes, and per-prefix per-peer attributes. Also, the computational load of validation of signatures in secure BGP is significantly higher in terms of the number of cryptographic operations that are required to validate an update message. However, as mentioned earlier, when an update leaves a BGPSEC community, BGPSEC signature attributes in the route are removed, so the storage overheads of BGPSEC are not seen by other BGP nodes.

It is important to note that initially BGPSEC community will mostly comprise of transit ASes, who are responsible for propagating BGP routes on behalf of other ASes. The study of Internet topology yields that approximately 15 percent of ASes are “transit” ASes. The remaining ASes behave as stub ASes that only originate routes and do not appear to transit routes for others. Such stub ASes can support a “light weight” simplex version of BGPSEC, where they provide BGPSEC signed originated routes to their upstream ASes, and nothing more [HUS11d].

While BGPSEC certainly provides an avenue for incremental deployment, it also means transitioning from 100% distributed control of today's BGPv4 based Internet to new layer of Internet control to be built for BGPSEC. The major concerns with BGPSEC pertain to scaling of this new architecture. This is a topic of much debate in IETF presently and the concern is whether enough has been done to ensure that BGPSEC will scale with the projected growth of the Internet routing table in the next few years. It is commonly recognized that today's Internet routing and addressing system is facing serious scaling problems [MEY07].

2.9 Other Contributions

In this section, we present and discuss several other contributions that have been made in the recent past.

Perlman [PER88] was among the first to recognize and study the problem of securing routing infrastructures. It was her work that demonstrated that any solution attempting to secure protocols like BGP must demonstrate Byzantine robustness; that is, if some nodes in the network start misbehaving, other nodes in the network should reach a decision on a particular message’s contents within a finite time period (termination). This decision should be the same among all these nodes (agreement), and the message should be the one sent by the source node (validity). It must be noted that existing solutions to date largely demonstrate only some facets of Byzantine robustness.

Before BGP version 4 was created, Kumar provided an analysis of security threats to interdomain routing protocols and proposed security mechanisms. These security mechanisms, adopted in the proposed IDRP [ISO92] can be viewed as an alternative to PKI. ISO's IDRP was designed as a superset of BGP and EGP and was initially tipped to replace BGP. However, due to popularity of IP, IDRP did not gain widespread traction. IDRP is a path vector protocol and uses an encrypted checksum for all routing messages sent between routers. The checksum uses an algorithm agreed by the two peers and is used to authenticate the messages. Authenticated timestamps and sequence numbers are used to protect against anti-replay attacks. However, the security mechanisms in the protocol did not account for prefix hijacking. Also, it advocated against encryption on per message basis due to the computation cost associated with cryptographic operations. Although IDRP highlighted important requirements for inter-domain routing security, it did not gain much adoption.

For the rest of this section, we categorize the contributions in five broad terms as follows:

- Cryptography based approach – the contribution makes use of cryptography in the proposed solution.
- Heuristics based approach – the contribution relies on one or more heuristics to improve BGP security.
- Probing approach – the contribution uses new probes (messages) to help ascertain correctness of BGP messages.

- Database or offline/parallel infrastructure – the contribution uses a database or an offline infrastructure.
- Attack types and incentives – the contributions discuss various attack types and their incentives and also propose approaches to combat these attacks.

2.9.1 Cryptography-based Schemes

Cheung [CHE97] proposed a symmetric-key based data authentication scheme whereby routing updates may be optimistically accepted with some form of validation data. The sender later releases the key needed to validate the routing updates. This requires time synchronization between parties. Being an optimistic link state verification scheme, it allows the use of an invalid routing update until the time that the update fails validation. Although this allows for faster convergence, there is a security tradeoff here. Also, in this scheme, when a misbehaving router is identified, it is isolated from the network. This aspect of the protocol itself can be misused as an attack mechanism to isolate perfectly valid routers from the network. Isolation of key routers in the Internet can cause major reachability issues in the Internet.

Heffernan [HEF98] defined TCP extension to enhance BGP security for BGP. It defined a new TCP option for carrying an MD5 digest in a TCP segment. This digest is based on keys known only to the peer nodes. This scheme hashed all or part of the TCP and BGP data message using one or more keys and addressed many of the problems of spoofing and hijacking present in TCP [BEL89], [GRE02].

Aiello et al. [AIE03] sought to assist with Origin Authentication of the incoming BGP UPDATE message, i.e., is the origin AS authorized to advertise an IP prefix? They propose using Origin Authentication Tags (OATs), which would carry the information needed to authenticate the owner of an IP prefix. The concern with this approach is that the IP prefix allocation scheme is complex and that multiple complex schemes would be needed to make it successful.

Subramanian et al. [SUB04] tries to detect inaccuracies in the data plane (the l*isten* part), but focuses also on control plane security (the w*hisper* part), and aims to provide an almost complete BGP security solution. However, the approach seems infeasible, as it tries to detect data plane anomalies by analyzing TCP flows, which can be millions per second on heavily trafficked routes.

Hu et al. [HU04] proposed SPV as a mechanism for securing BGP that replaces the computationally expensive asymmetric cryptography, as used in S-BGP, with a signature scheme based on symmetric cryptography. SPV protects against AS-PATH truncation and modification attacks. The protocol defines three concepts to secure AS-PATH. First, it includes private keys within the updates. Second, it relies on hop-by-hop authentication to check if no AS was inserted onto the path. Last, it limits the options of an attacker can have to attack the network. The SPV protocol offers a balance between computational complexity and strong route authenticity guarantees. However, Raghavan et al. [RAG07] described design flaws that can lead to successful attacks on the SPV protocol, such as Multi-path modification and truncation attacks. Because of these vulnerabilities, SPV protocol doesn't offer the same security guarantees as the S-BGP.

Aiello et al. [AIE06] proposed to depreciate the cost of cryptographic signature verifications by considering the reference locality of BGP announcements [BUT06]. The authors reviewed the Route Views data archive [UORV] and noticed that paths are generally stable, and the number of new paths grows fairly slowly. Based on this, they suggested a new path authentication mechanism (as compared to S-BGP). This mechanism maintains a similar level of security while substantially reducing the number of signature validations required. However, there is a very significant increase in the bandwidth requirements because of the large cryptographic proof systems that are distributed.

Zhang et al. [ZHA09] proposed a new architecture to secure IP prefix ownership. The proposal, named, HC-BGP, uses one-way hash chain and regular public/private key pairs to ensure prefix ownership certificates. The authors also provide a partial ordering algorithm to prevent any malicious network from tampering the messages. HC-BGP requires verification only when the origin changes and it relies on the existing hop-by-hop trust relationship.

Yin et al. [YIN10] proposed a keychain-based signature scheme called *KC-x*. The keys used for signature generation and verification form a chain by themselves, resulting in a strong link between signatures. *KC-x* builds a chain of key authorization along an AS-PATH and has the flexibility of using different signature algorithms, which can even co-exist in a hybrid deployment.

[Li11] proposed TBGP, a trusted BGP scheme which aims to achieve high authenticity of Internet routing with a simple and lightweight attestation mechanism. TBGP proposes a set of route update and withdrawal rules that,can guarantee the authenticity and integrity of route information. Through this, TBGP builds a transitive trust relationship among all routers on a routing path.

2.9.2 Heuristics-based Schemes

Bradley et al. [BRA98] proposed WATCHERS protocol to detect and react to routers that drop or misroute packets. WATCHERS tracks the packets flow in each node (both incoming and outgoing), and detects routers that violate the conservation principle which states that all data bytes sent into a node are expected to exit the node, unless they are destined for this node. Routers participating in WATCHERS count packets in various categories, share the packet counts with other participating routers, and perform diagnostics on the data to determine if a router was misbehaving in the network. This protocol is mainly targeted for intra-AS routers. From inter-AS point of view, it is difficult to imagine different ASes divulging characteristics of traffic flow, especially when the result of the protocol may be some type of penalty levied against a router.

Zhao et al. [ZHA01a] suggested using change in the origin AS as one of the heuristics which determine whether the BGP route is valid or not. If originating AS for a particular route changes, the validity of the route can be questioned. RFC 1930 [HAW96] recommends that an IP prefix should be originated from a single AS with few exceptions. A study by Zhao [ZHA01b] showed that show, that Multiple Origin Autonomous System (MOAS) is a common phenomenon, caused by either operational practice such as support for multi-homing, or by misconfiguration or software defects. A MOAS conflict occurs if an IP address prefix appears to originate from more than one AS. Blind acceptance of MOAS conflicts is potentially dangerous as it opens the door for traffic hijacking. In this contribution, Zhao defined a new BGP Community, termed "MOAS List", to distinguish valid MOAS conflicts from invalid ones. However, because the community attribute is optional and transitive, routers can drop this information without causing an error.

Considering the importance of DNS to network architecture, Wang et al. [WAN03a] proposed a scheme to protect BGP routes to top-level DNS servers from modification. It employs path filtering based on heuristics, such as those used for MOAS detection [ZHA01a]. The path filtering scheme is seemingly practical for this particular application because routes to DNS servers are found to be stable.

Kruegel et al. [KRU03] propose evaluation of common BGP configurations and AS behavior and then use metrics to identify bogus announcements (e.g., strange aggregation and tracking of historical associations between prefixes and ASes). For example, the proposal observes ownership over time. If this ownership changes in a new route (a new AS begins to announce the address, or a new MOAS occurs), it is considered to be malicious and is flagged. They demonstrate the number of false positives is relatively small, on the order of twenty per day compared to over five million UPDATE messages processed per day. However, the prefix ownership lists are static in this model and need to be rebuilt if a topology change occurs in the network.

Wan et al. [WAN05] proposed Pretty Secure BGP (psBGP) architecture which considers an address origin authentication scheme within a larger comprehensive BGP security architecture. The basic assertion in this proposal is that it is not practical to manage addresses through a centralized PKI. For origin authentication purposes, each AS rates every other AS with a metric to indicate trustworthiness of the foreign AS. Each AS creates a Prefix Assertion List (PAL) which contains address ownership assertions of the local ASes and its peers. The PALs of peers are checked for consistency whenever an origin claim is being validated. However, this form of weak origin authentication does not protect against two or more colluding ASes from attacking the network with false origins.

Karlin et al. [KAR06] proposed an alerting system, called Pretty Good BGP (PGBGP). With this scheme, historical routing data is maintained and used to determine what routes to prefixes should be considered normal. If an incoming route conflicts with historical route, it is flagged as suspicious for 24 hours. This time limit of 24 hours is based on work by Mahajan et al. [MAH02] that shows most misconfigurations and hijack attempts last for less than this amount of time. The route is also avoided while it remains suspicious. While this scheme may offer good protection for ASes against prefix hijacking attacks, it can impact convergence in the network if it incorrectly marks a new, better route as suspicious.

Blazakis et al. [BLA06] proposed a scheme to measure the difference between the current path and the new path in terms of a quantifiable metric, called edit distance. Edit distance is defined as the number of operations needed to change one string to another. This can be used when comparing AS-PATHs. Edit distance is equal to the number of ASes that are different in the AS-PATH. For example, if two ASes are different, the edit distance will be equal to two. The reliability of the routing update decreases when edit distance increases. Also, changes in the path are weighted differently, depending on the position of the AS that has changed. While this proposal gives a hint for a discrepancy, it is not necessarily reliable or accurate in all cases.

2.9.3 Probing Schemes

Qiu et al. [QIU06a] proposed that routers, when they detect an origin change, send a probe to find the true origin and then notify the true origin if a discrepancy is found. This scheme is capable of running on current router hardware, instead of requiring new hardware. Also, it does not require cryptography. While it can handle origin change misconfiguration, it does not handle malicious nodes well.

Hu et al. [HU07] proposed using AS/router properties as a finger-print to validate access to an IP prefix. Examples of such properties include: operating system, router services, and ports in use.

When two different routes are received, these properties can be compared to detect an invalid path. The system sends out probing messages to obtain a fingerprint with the hope that only valid routers originating the prefix will respond to the probing messages. The concern with this approach is that some of the probe techniques may be impossible due to firewalls in the path.

Zhang et al. [ZHA10] proposed *iSPY*, which is a prefix-owner-based IP prefix hijacking detection system. *iSPY* relies on the rich connectivity of the ASes in the Internet to detect that IP prefix hijacking causes an outage with a signature which is distinct from typical link failures in the network. *iSPY* continuously monitors network reachability from external transit networks to its own network through lightweight probing and scans for the hijacking signature as the trigger for hijacking alarms.

2.9.4 Database or Offline/Parallel Infrastructure

Reynolds et al. [REY06] suggested using a trusted platform to monitor BGP router, thereby, creating a security plane to query a route update. It allows for parallel infrastructure to secure anomalies in BGP UPDATE messages. However, adding systems to monitor routers or validate routes requires dedicated network resources and creates more maintenance work for AS administrators.

Wendlandt et al. [WEN06] proposed Availability-Centric Routing (ACR), which suggests that a database of route history and performance, called a route repository, be used to determine the optimal path for a source, destination pair. ACR views the route updates as candidates to be evaluated against a superset of paths which could be provided offline, at a cost, by an availability provider (AP). A transit AS, acting as an AP, maintains database of path characteristics and helps provide multiple candidate paths. The service would allow data to be sent along these paths using key points in the network, called deflection points, using tunneling. This proposal seeks to guarantee that the status of forwarding table is consistent with the advertised BGP routing information. However, the proposal is a dramatic departure from current architecture of BGP deployments.

Lad et al. [LAD06] proposed Prefix Hijack Alert System (PHAS) which uses routing database containing BGP route updates to identify IP prefix hijack events. This database can be created based on the number of different BGP monitoring projects, e.g. Route Views [UORV]. In the case of IP prefix being hijacked, the operator is notified via email alerts. The system is dynamic in that it automatically adjusts the time needed to make a determination and the number of alerts to be sent out.

Liao et al. [LIA10] study the diverse and evolving commercial agreements that ISPs (ASes) enter into for peering relationships to be able to identify safe and robust routing policies. The authors investigate different routing policies which can be devised to accommodate complex mutual transit agreements. They propose policy guidelines to allow mutual transit agreements and guarantee routing safety and robustness as long as the AS graph satisfies a corresponding set of precise topological constraints.

Wang et al. [WAN10] proposed to have each router build a database of IP prefixes and their rightful owners so that any BGP UPDATE message can be verified for rightful origination of the IP prefix. To detect AS-PATH manipulation, the authors proposed to extend the BGP to contain an attribute as-path-type used to carry the type of a route.

2.9.5 Attack Types and Incentives

Ballani et al. [BAL07] detailed the impact of IP address prefix hijacking by establishing a methodology for prefix interception and estimating the fraction of traffic to any prefix that can be intercepted in the Internet. The methodology showed that that ASes higher up in the routing hierarchy can hijack a significant amount of traffic to any prefix, including popular prefixes.

Goldberg et al. [GOL08] studied conditions which can encourage ASes to announce BGP routes that are different from the paths that data packets traverse in the data plane. In general, ASes seek the best possible outgoing path for their traffic. In practice, AS is also interested in attracting incoming traffic (*e.g.,* because other ASes pay it to carry their traffic). The study looks at combinations of BGP enhancements and restrictions on routing policies which can ensure that ASes have no incentive to lie about their data-plane paths. Using game-theoretic analysis, it is demonstrated that protocols like S-BGP alone are insufficient, but that S-BGP does suffice if coupled with additional (quite unrealistic) restrictions on routing policies.

McArthur et al. [MCA09] exposed a new class of stealthy prefix hijacking attacks that are harder to detect. This kind of attack methodology affects a relatively small fraction of ASes while the victim continues to receive traffic from the majority of ASes (raising no alarms). The basic idea is to tune the length of the invalid paths so that they are only appealing to a smaller fraction of ASes. Using Route-Views [UORV] topologies, the authors present upper bounds on the impact of stealthy prefix hijacking attacks. The authors also propose a defense mechanism which uses a combination of existing detection methods to successfully detect a stealthy prefix hijacking attack.

Goldberg et al. [GOL10] studied the effectiveness of existing major BGP security techniques (origin authentication, soBGP, S-BGP, and data-plane verification) to prevent traffic-attraction attacks. Through simulations, the authors demonstrate that even in the presence of S-BGP or data plane verification, an attacker can maximize the traffic he attracts by widely announcing a short, but valid, AS-PATH. The authors also studied different kinds of attacks and determined that determining most damaging attack strategy is NP-hard. It can be concluded that mechanisms that police export policies (e.g., defensive filtering) are crucial, even if S-BGP is fully deployed.

Shue et al. [SHU12] studied malicious activity happening around the Internet and explored whether some large profile ASes are so-called safe havens for these malicious attacks. The authors studied blacklists of IP addresses, local spam data and DNS resolutions of the IP addresses in the blacklists and established that some ASes have more than 80% of their IP addresses blacklisted. Also, several ASs regularly peer with ASs associated with significant malicious activity. The authors also studied properties of malicious ASes which distinguish them from normal ASes.

2.10 Conclusion

In this chapter, we presented a brief overview of BGP and its operation. We also discussed different security techniques (cryptography, protection of BGP session, defensive filtering, and routing registries) to secure various aspects of BGP. We presented a detailed overview of the four most comprehensive approaches to BGP security: S-BGP, soBGP, IRV and BGPSEC. We also discussed a number of other schemes that have been proposed in the last several years. Finally, we also discussed different attack types and incentives for attacking the Internet

Chapter 3

Extensions to BGP and Data-plane Security

3.1 Introduction

The act of forging source addresses in IP packets, also known as IP address spoofing, remains an effective mechanism to launch DoS attacks on the Internet [RIC00],[NAR02]. The idea behind source IP address spoofing is to craft a malicious attack packet and send it to the network with a source IP address that belongs to some other AS in the network. This allows the attacking node/AS to pose as another AS and also hide its actual identity and location, making it difficult to detect the actual attacker and to protect against it. As a result, such types of attacks are generally immune to detection techniques that rely on source address-based filtering.

There are several reasons why source IP address spoofing remains a popular method to launch attacks in the Internet [MOO06]. First, when an attack is launched using source IP address spoofing, it is difficult to differentiate attack traffic from legitimate traffic. The AS whose IP address has been hijacked may well be sending legitimate traffic at the same time as attack traffic is being sent from its IP address. Second, although the attack appears to be coming from a particular victim AS (whose source IP address has been hijacked), it can take substantial amount of time and resources to determine that the host itself is a victim and that the true attacker still needs to be located [BEL03b],[SAV00],[SNO01]. Finally, forging of source IP addresses allows the attacker to pose as a valid AS on the other end of a transaction and launch popular man-in-the-middle attacks, such as variants of TCP hijack and DNS poisoning attacks [RAM10],[STE03]. Similarly, IP spoofing can be used to launch reflector-based attacks whereby an attacker uses some victim's IP address to contact a number of hosts, resulting in the victim being flooded by replies from all these hosts [PAX01].

Many solutions have been proposed to detect IP spoofing. Most of them are based on filtering packets based on the IP source address and the incoming interface. The premise is that if the source IP address of the packet is not *expected* to be received on the incoming interface then the packet is dropped. We shall discuss inter-domain packet filters (IDPF) scheme in the next section.

3.2 Inter-domain Packet Filters (IDPF)

The route based packet filter proposed by Park and Lee [PAR01] relies on the assumption that if a single-path routing scheme is assumed, there is exactly one single path *p(s, d)* between source node *s* and destination node *d*. Therefore, it is acceptable to discard any incoming packet with source address *s* and destination address *d* not in *p(s, d)*. However, it is not possible to create accurate route-based packet filters at a given AS without this AS possessing the knowledge of entire global routing decisions made by all other ASes in the network. This is impossible with the current BGP-based Internet routing infrastructure. As discussed earlier, BGP is a *policy-based* routing protocol, whereby locally defined policies at an AS influence both the selection as well as the propagation of the best route to reach a destination. These policies are typically closely guarded by individual ASes as they tie into revenue aspects of the ASes. Given this, it is virtually impossible for an AS to acquire the complete knowledge of routing decisions made by the other entire ASes. This significantly hampers the adoption of route-based packet filters.

The IDPF architecture takes advantage of the fact that while network connectivity may imply a large number of potential paths between source and destination domains, commercial relationships between ASes act to restrict to a much smaller set the number of *feasible* paths that can be used to carry traffic from the source to the destination [DUA08]. The commercial relationships between ASes can be summarized as following [LIA10], [HUS99]:

provider-customer: In this arrangement, a customer AS pays the provider AS to carry its traffic to the rest of the Internet. This is very commonly observed arrangement in the Internet today where smaller ASes are customers of much larger provider ASes.

p*eer-peer*: In this arrangement, ASes of roughly same size agree to carry traffic from each other (and their customers).

s*ibling-sibling*: In this arrangement, two ASes provide mutual transit service to each other.

Table 3-1: Route export rules at an AS [DUA08]

Export rules		***r1***	***r2***	***r3***	***r4***
Export routes to		provider	customer	peer	sibling
Learned from	provider	no	yes	no	yes
	customer	yes	yes	yes	yes
	peer	no	yes	no	yes
	sibling	yes	yes	yes	yes
Own routes		yes	yes	yes	yes

Table 3-1 captures the rules for route export for ASes with different relationships. In this table, rules r1-r4 dictate the export policies typically used by an AS to announce routes to providers, customers, peers, and siblings, respectively. As an example, export rule r1 states that "an AS will announce routes to its own networks, and routes learned from customers and siblings to a provider, but it will not announce routes learned from other providers and peers to the provider" [DUA08]. These export rules can be handily used to restrict the number of possible paths between each pair of ASes. Also, these export rules are the cornerstone of IDPF framework.

The main idea behind the IDPF framework is to only allow data packets from *feasible* upstream neighbors to pass and discard all other packets. Such filtering needs to make sure that it will not discard packets with valid source addresses. IDPFs are constructed from the information implicit in BGP route updates. When an AS receives a packet from an incoming interface, it checks if the source IP address has been advertised through this interface. The packet is discarded if the check is negative. A key feature of the scheme is that it does not require global routing information.

3.3 Performance of IDPF

We define performance of IDPF as the ability of the network to prevent IP address spoofing attacks in the network. We developed our own simulation system to study the performance of IDPF scheme in the presence of both uncompromised and malicious ASes (see Section 4.4 for details). The simulation models each AS as a node in the system and allows for a random topology with the option of the user to specify average degree of connectivity for each AS. It models propagation of BGP UPDATE messages in the network. It models IP address spoofing attacks by randomly (using uniform probability distribution) selecting ASes which send IP packets with spoofed source IP addresses (IP addresses belonging to different ASes). The simulation models IDPF-enabled node behavior by maintaining a database which contains a mapping between source IP addresses and interfaces on which routes for these source IP addresses were learnt. The simulation will drop IP packets if they are received on interfaces which are different from the interface in the database. All such drops are counted towards success of IDPF scheme. Any spoofed IP packets which do not get dropped and are delivered to the destination address are considered as towards the success of the attack.

For current analysis, the simulation system consisted of 2000 ASes in a random topology configuration and with shortest paths being the criteria of choice for selecting the best route from each AS to other ASes. We ensured that the simulation system yielded results similar to those presented in [DUA08] for IDPF. This gave us confidence in the reliability of the results from the simulation system. The simulation was repeated five times to compute 95% confidence intervals. The 95% confidence interval means that 95% of the simulation results fall within the interval. Throughout this thesis, the confidence interval is computed based on five independent runs. It was observed that more than 95% of the results were within the calculated confidence interval for each experiment. The confidence intervals themselves are shown on each figure.

Figure 3-1 demonstrates the success of IDPF scheme against IP spoofing attacks. For example, when 50% of the ASes have IDPF enabled, approximately 40% of the IP spoofing attacks can be stopped.

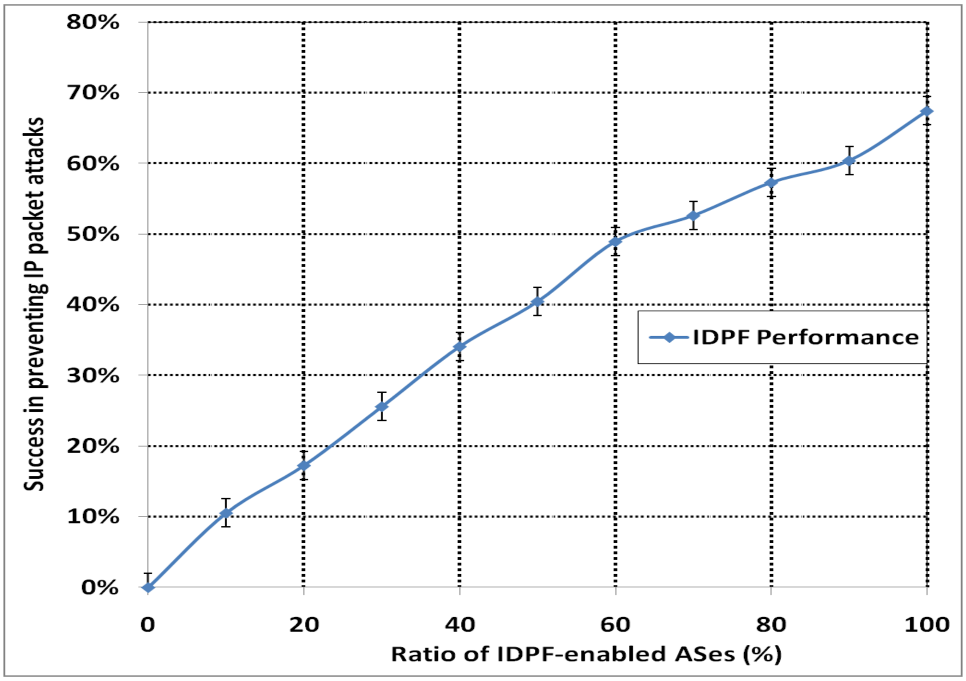

Figure 3-1: IDPF Performance.

It is important to note that when IDPF is deployed on all the nodes (100% IDPF), the success of the scheme is approximately 67%. Interestingly, it is not 100%. This can be explained using the example in Figure 3-2. In this example, IDPF is deployed on all ASes. AS *G* advertises a BGP route UPDATE message, which reaches AS *A* via ASes *E* and *C*. If AS *E* decides to spoof AS *G*'s IP address and send an IP packet with *G*'s IP address, this packet will be accepted by ASes *C* and *A*. This is because the IP packet is following the same AS path as the BGP UPDATE message. The interface checks will match for both IP packet and BGP UPDATE message. In such scenario(s), IP spoofing attack will succeed even with 100% IDPF deployment. From our simulation results, when all nodes deploy IDPF, an average of 67% of the attacks can be prevented. The rest of the attacks are successful.

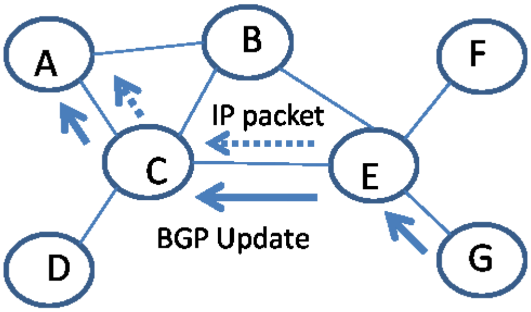

Figure 3-2: An example of a scenario where IDPF cannot protect against IP spoofing attacks.

3.4 Impact of BGP Prefix Hijacking on IDPF

The IDPF scheme assumes that BGP routing updates are secure and hence trustworthy. As discussed earlier, IDPF filters are created based directly on BGP UPDATE messages. Hence, if the BGP UPDATE messages can be manipulated to carry incorrect routing information, performance of IDPF scheme will suffer. For example, if malicious ASes are introduced in the network who can hijack another AS's IP prefix and announce themselves as owners of those IP prefixes, it can lead to creation of incorrect IDPF filters. This will directly lead to degradation of IDPF filter performance. We repeated simulation analysis with one AS in the network participating in IP prefix hijacking attacks. The selection of AS was random, using uniform probability distribution model.

Figure 3-3 demonstrates that IDPF performance degrades in the presence of false BGP updates in the network. As an example, when IDPF is enabled on all nodes (100% deployment), approximately 28% of IP attacks can be prevented when one AS is advertising bogus BGP updates in the network. Without any bogus BGP UPDATE messages, the success rate is 67%. These results demonstrate that there is a strong incentive for attackers to combine both control plane and data plane attacks to divert traffic in the network.

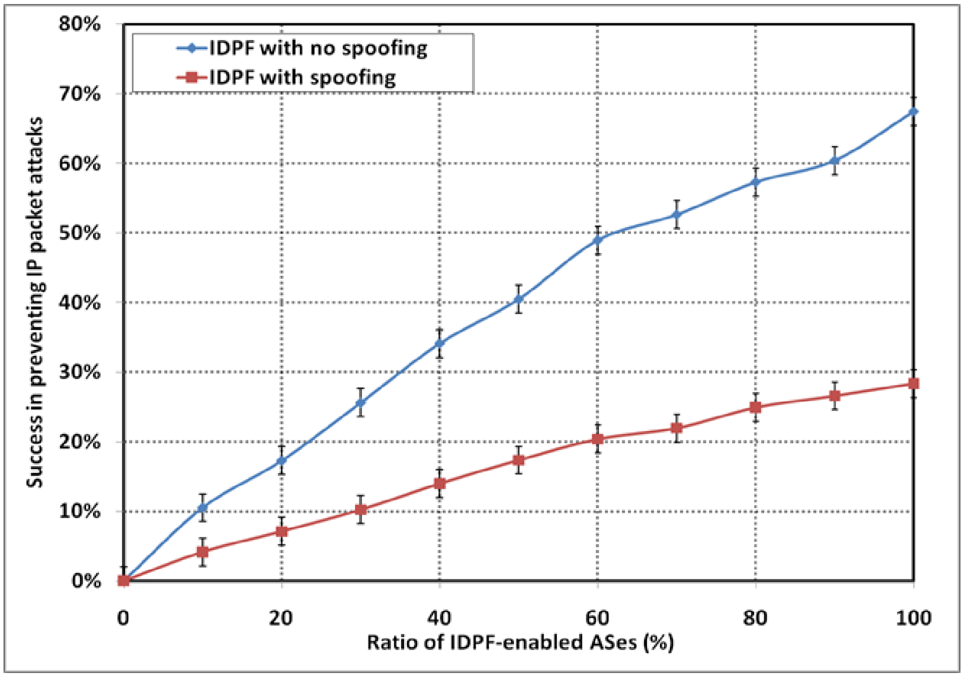

Figure 3-3: Degradation of IDPF Performance in presence of bogus BGP UPDATE messages.

The results in Figure 3-3 reinforce the need to address IP spoofing attacks both on data as well as control planes. The decline in IDPF performance can be arrested by deploying a mechanism to secure BGP such that bogus UPDATE messages are not allowed to traverse the network. There needs to be a way to ensure that BGP UPDATE message is advertising a prefix that is rightfully owned by the AS originating the UPDATE message. Also, there needs to be a way to ensure that the BGP UPDATE message actually traverses the AS-PATH listed in the BGP UPDATE message.

3.5 Extensions in BGP for Secure Routing

There are a number of practical and a number of more fundamental questions relating to securing BGP. The first is a practical question relating to the inevitable design trade-off between the level of security and the performance overheads of processing security credentials associated with

BGP UPDATE messages. The question concerns what aspects of securing BGP should be considered essential and what is considered to be desirable, but not essential. It is not entirely known as to what aspects of BGP performance and load are critical for the robust operation of network applications and what are not so critical. With such considerations, it is important that any solution to secure BGP should try and minimize impact on current performance of BGP and should be incrementally deployable. In order to add security, we propose the following concepts and extensions in BGP [ISR09b], at a very high-level:

1. Add ability in BGP node to validate that the AS number claiming to originate an IP address prefix (as derived from the AS-PATH attribute of the BGP route) is in fact authorized. Whenever an UPDATE message is received, the receiving AS will try to determine whether the originator AS is authorized to originate the IP prefix.
2. Add ability in BGP AS to perform complete (or partial) attestation of the AS-PATH contained in the BGP UPDATE message. Whenever an UPDATE message is received, the receiving AS will be able to confirm complete (or partial) validity of the AS-PATH.
3. Add intelligence in BGP selection algorithm to consider results of the above two steps. We add intelligence in BGP selection algorithm to consider the results of both 1 and 2. The idea is to get separate results for both 1 and 2 above and then use a simple mathematical equation to get an overall result. The overall result, termed as “Route Credibility Score” (RCS), can be derived as follows:

$$RCS = min\,(X\,,Y) \tag{3.1}$$

Where:

$X = \text{Score (IP address prefix origination)}$

$Y = \text{Score (AS} - \text{PATH validation)}$

Both *X* and *Y* can be defined as follows:

Table 3-2: Different possible values for X and Y

	Value	**Meaning**
X (or *Y*)	2	Valid
	1	Unknown
	0	Invalid

In essence, if there is a reliable scheme to determine IP address prefix origination and if this scheme confirms that the incoming prefix was originated by the rightful owner of the prefix, *X* would be assigned a value of 2 (Valid). If this scheme confirms that the originator of the prefix is in fact not the rightful owner of the IP address prefix, *X* would be assigned a value of 0 (Invalid). If there is no reliable way to determine the originator of the address prefix, *X* would be assigned a value of 1 (Unknown). Similarly, if there is a reliable scheme available to attest validity of AS-PATH and if this scheme confirms that the incoming AS-PATH is in fact valid, *Y* would be assigned a value of 2 (Valid). If the scheme confirms that AS-PATH is not valid, *Y* would be assigned a value of 0 (Invalid). If there is no reliable way to attest validity of AS-PATH, *Y* would be assigned a value of 1 (Unknown).

RCS would be assigned the minimum of *X* and *Y*. An "Invalid" *X* or *Y* value will give minimum value to *RCS*, thereby making it less preferable to another route with a higher *RCS* value. This is because, under, no circumstances, should a route with "Invalid" value of *X* or *Y* be accepted, as we know that its origin or AS-PATH is invalid. We propose to change BGP selection algorithm to consider *RCS* value for each incoming route. The standard BGP selection algorithm is given in

Table 3-3. We propose to add a check for RCS value as step 1 of the selection algorithm, as in Table 3-4.

Table 3-3: BGP route selection algorithm

Step	Criterion
1	Highest Local Preference
2	Lowest AS-PATH length
3	Lowest origin type
4	Lowest MED (with same next-hop)
5	eBGP-learned over iBGP-learned
6	Lowest IGP path cost to egress router
7	Lowest router ID of BGP speaker

Table 3-4: Proposed new BGP route selection algorithm

Step	Criterion
1	Highest non-zero RCS value, reject RCS value of 0
2	Highest Local Preference
3	Lowest AS-PATH length
4	Lowest origin type
5	Lowest MED (with same next-hop)
6	eBGP-learned over iBGP-learned
7	Lowest IGP path cost to egress router
8	Lowest router ID of BGP speaker

In order of preference, the proposed algorithm will always prefer routes with *X* and *Y* values which are “Valid”, followed by “Unknown”. It will reject routes with *RCS* value of 0 as they are definitely suspicious or incorrect. If two routes have the same non-zero *RCS* value, then the next steps in the standard BGP selection algorithm will apply.

In the next section, we present few schemes which can be used to determine *X* and *Y* values, which are needed to determine value of *RCS*.

3.5.1 Potential Route Validation Schemes

Resource Public Key Infrastructure (RPKI) [LEP11b] can be accessed by each AS in the network to query the validity of a received BGP update message. Upon receipt of a BGP update message, the AS performs a RPKI database lookup to determine the owner AS for the received prefix. The result can then be compared with the AS contained in the BGP update message to determine if there is a match. If the RPKI database is not available, or if the prefix is not listed in the database, the result for *X* can be considered as an "Unknown". If the prefix exists in the database and a "match" is found, the result can be considered as "Valid". If the database lookup concludes that the prefix should have been originated by an AS different from what is received in the update message, the result can be considered as "Invalid".

The RPKI can be extended to store information about external peering information for each AS. The database would also store who the direct peer ASes are for each AS in the network. Whenever a BGP update message query is made in the RPKI database, the retrieved information can also provide this topology information. Using this topology information, a *feasible* path can be created in each node's local database for the ASes contained in the BGP update message. In this context, a feasible path means a topological path between two ASes. Each BGP node can determine whether the received AS-PATH information matches the *feasible* path. If *feasible* path information is not available, the result for *Y* can be considered as "Unknown". If the received AS-PATH information matches the feasible path, the result for *Y* can be considered as "Valid". If the AS-PATH information does not match the feasible path, the result can be considered "Invalid".

Another approach which can be used to validate the origin of the prefix is presented in [GOO03]. In this scheme, a server is dedicated on per AS-level to respond to queries from BGP speakers in other ASs. At a high-level, this server is capable of carrying AS specific routing policy information, BGP community information, local topology information and received/sent routing updates. The AS server is designated by each AS and is made to be reachable by all other ASs in the system. Whenever a routing update is received at a BGP node, the node can query the AS server for the ASs listed in the update message to determine if the AS claiming to be originating the prefix is in fact authorized to do so. If the AS server is not available, the result for *X* can be considered as “Unknown”. If the AS is in fact authorized to announce the prefix, the result can be considered as “Valid”. If the server query concludes that the prefix should have been originated by an AS different from what is received in the update message, the result can be considered as “Invalid”. Since the server is designed to store route updates sent to each neighboring AS, this information can be used to validate AS-PATH information. When a routing update is received by a BGP node, the server for each AS listed in the AS-PATH can be contacted to confirm/deny whether the route update was in fact sent by a BGP speaker in that particular AS. Again, if the AS server is not available, the result for *Y* can be considered as “Unknown”. If each AS server confirms that the route update did in fact traverse its respective AS, the result can be considered as “Valid”. If any of the AS server responds with negative, the result can be considered as “Invalid”.

A simpler approach to validate AS-PATH would to infer the feasibility of the AS-PATH based on a network topology built from received BGP UPDATE messages. In steady state, a BGP node would have enough route updates accepted to build a topological view of the network and AS-PATHs. In this state, if a BGP update is received with an AS-PATH which is not among the feasible paths in the local topology, the BGP update is not immediately accepted. The concern with this approach is that it may never accept valid BGP update messages. To address this, the

BGP node can be configured to start accepting such messages if they are seen on multiple interfaces. It is acknowledged that this may not be the most desirable approach in terms of security strength and/or BGP performance; it is a viable option considering the overhead associated with other techniques presented in the literature.

3.5.2 An Example Scenario

As discussed earlier, the proposed solution provides security both on the control plane as well as on the data plane making the route selection process for BGP secure on both levels. We will demonstrate the process by a small topology example. In our topology, every node presents the speaker node for every AS. We assume that the security solution is implemented in the network and all the nodes are in converged state; in the topology, seven ASes are connected with each other and all are connected to the repository server for the retrieval of information for the authentication process; the database can be created on the router memory but in this scenario we'll take the server case.

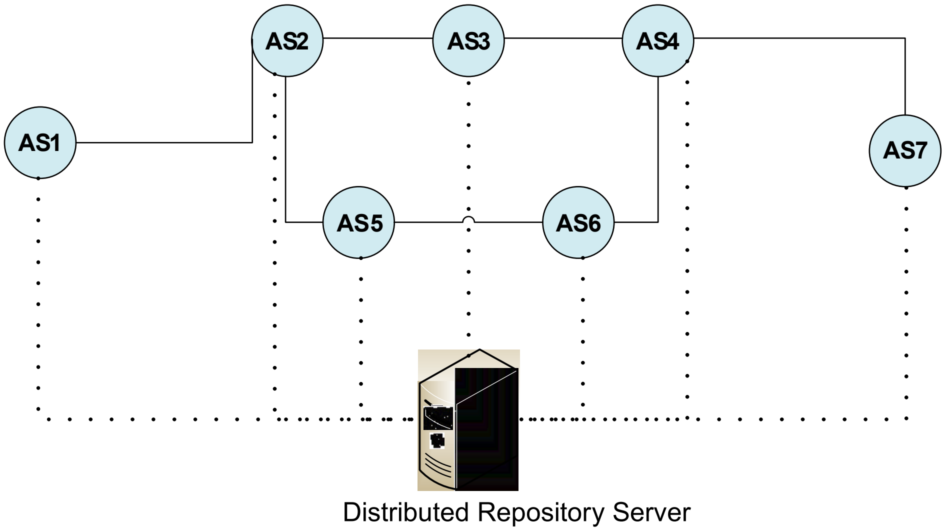

Figure 3-4: Network topology for the example scenario.

Table 3-5: Prefix ownership and direct neighbor relationship information

Node	AS Prefix	Neighbors
AS1	192.168.1.0/24	AS2
AS2	192.168.2.0/24	AS1, AS3, AS5
AS3	192.168.3.0/24	AS2, AS4
AS4	192.168.4.0/24	AS7, AS6, AS3
AS5	192.168.5.0/24	AS2, AS6
AS6	192.168.6.0/24	AS4, AS5
AS7	192.168.7.0/24	AS4

We observe the scenario of AS-PATH manipulation on the route from AS1 to AS4. We are interested in two attributes carried in the UPDATE message, the AS-PATH and the IP prefix being announced. Let’s assume that the following AS-PATH and IP Prefix are received by AS2.

AS-PATH: AS1 ► AS2

AS’s IP Prefix: 192.168.2.0/24

An attack is initiated from AS2 who manipulates AS-PATH and forwards the UPDATE message to AS3 with following contents:

AS-PATH: AS1 ► AS2 ► AS6 ► AS3 ► AS4

IP Prefix: 192.168.2.0/24

When this UPDATE message reaches AS3, the proposed solution will perform following three steps:

1. Contact the repository server to retrieve the IP prefix ownership and AS neighbor relationship information.
2. Compare the information retrieved from the server against the information received in the UPDATE message.
3. Calculate *RCS* and then decide to whether accept or drop the UPDATE message.

In this scenario, since the AS Prefix values match, $X = 2$. However, the AS-PATH contains AS6 which does not match against the neighbor values of AS3 (its neighbors are AS2 and AS4). Hence, $Y = 0$. The RCS is determined as the minimum of X and Y.

$$RCS = Min(2,0) = 0$$

The proposed enhancement to BGP selection algorithm will force this UPDATE message to be rightfully dropped.

A similar example can also be created to illustrate how the proposed scheme prevents BGP UPDATE messages with hijacked IP prefixes can be prevented from traversing the network.

3.6 Impact of Proposed Extensions on IDPF Performance

In this section we aim to study the impact of adopting the proposed extensions on the performance of the IDPF filters. We repeated the simulation model with an increasing number of ASes adopting the security measures discussed above. Figure 3-5 shows the success of the network in preventing IP packets attacks when an increasing number of ASes adopt proposed security measures. The simulations were conducted for different ratios of IDPF-enabled ASes ((20%, 60% and 100%). It can be observed that as more ASes adopt the security measures, the success ratio for the network improves. With 100% IDPF adoption and all ASes adopting security measures, the performance of the network is the same as in Figure 3-1 (with 100% IDPF-enabled ASes). This is because when all ASes adopt security measures, then there is no chance that a bogus BGP UPDATE message will cause creation of incorrect IDPF filters. There is a strong incentive for ASes to adopt both IDPF filters as well as BGP security mechanisms to mitigate against IP spoofing attacks.

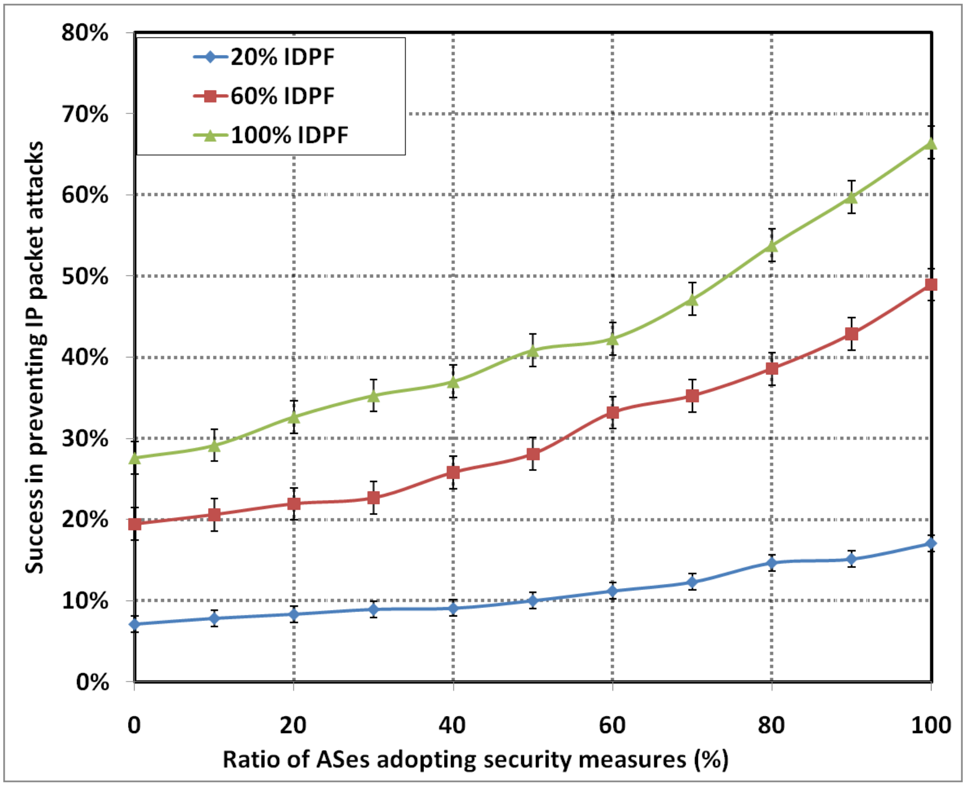

Figure 3-5: Success in preventing IP attacks with ASes adopting proposed security measures.

3.7 Conclusion

IDPF was proposed as a solution to mitigate against data-plane attacks. The packet filters are created based on BGP UPDATE messages as they are received by ASes. In this chapter, we demonstrated that the performance of IDPF filters suffers heavily when BGP messages are spoofed. Bogus BGP UPDATE messages will lead to successful data-plane attacks. This analysis confirmed that securing BGP is not only important from control plane perspective but also from data plane point of view. As a solution, we propose simple extensions to BGP selection algorithm. We assume that there exists a separate mechanism to validate the incoming BGP UPDATE messages. Once the validity of the message has been ascertained, the extensions to BGP selection algorithm allow BGP to either accept or reject these messages, depending on whether they are valid or not. If used properly, this will prevent creation of invalid IDPF filters. We studied the impact of these extensions on IDPF filters in presence of malicious BGP UPDATE messages and confirmed that we are able to restore the effectiveness of IDPF in combating data plane attacks in the presence of both control and data plane attacks.

Chapter 4

Adding Trust to Secure BGP

4.1 Introduction

Validating digital signatures is a computationally expensive operation. Nevertheless, Secure Border Gateway Protocol (S-BGP) mandates that upon reception of an update, an S-BGP speaker must verify nested signatures of all ASes in the traversed path; and the router should verify the Address Attestation to check if the source has the right to announce the address prefix. S-BGP requires several digital signatures in each UPDATE, and as a result has a high CPU overhead for verifying UPDATE messages. The computation requirements of the protocol dramatically slow down the network convergence. We propose a new approach, called Credible BGP (C-BGP), that reduces the burden of validating the AS-PATH and the address prefix origination. We define a control layer of *trusted* ASes that is comprised of major Autonomous Systems (ASes) in the network. In this environment, a *non-trusted* AS has to verify only the signatures of intermediate ASes between itself and the last *trusted* AS in the AS-PATH. Similarly, the address prefix is validated only if it was not previously validated by a *trusted* AS. Using analytical and simulation studies, we measured performance metrics of the new protocol. We show that even with small ratio of *trusted* ASes, C-BGP can significantly reduce the number of verifications required to validate AS-PATH and IP prefixes.

4.2 Performance and Operational Issues in S-BGP

Even though S-BGP provides comprehensive security to BGP, there are a number of issues associated with its deployment. The sheer volume of security related data being and signatures required makes the validation costly in terms of computational load and protocol data exchange. Due to “onion-style” validation adopted in S-BGP, the total number of signature verification grows as BGP UPDATE message traverses AS boundaries.

The excessive computational load has the ability to impact BGP network convergence which can cause instability and degraded network performance.

Earlier, in the design phases of S-BGP, the BGP traffic analysis yielded that busy hour rate for a router with 30 peers would be about 5 UPDATE messages per second [KEN03b]. Also, on average, each UPDATE message would contain about 3.7 Route Attestations (RAs). This amounted to a total of approximately 18 signature verifications per second.

However, latest data shows that each UPDATE message traverses on average 4 ASes and the incoming rate may be as high as 300 UPDATE messages per sec in an attack scenario [HUS09b]. In the worst case scenario whereby a router restarts and is in receipt of such high rate of UPDATE messages, the router software would need to execute 1200 signature verifications per second. This would be very challenging to sustain in an attack scenarios without seriously compromising the normal functionality of the BGP AS.

Furthermore, when an S-BGP UPDATE message is to be forwarded to multiple peers, a different signature per peer is needed in each UPDATE message. The RA specifies a unique AS number of the peer to which the UPDATE message is sent. This adds to the computational requirement of each AS generating the UPDATE message. Initialization/reboot of a BGP router also results in a surge in UPDATE processing. From recent research, it is evident that while the "diameter" of the Internet has mostly remained the same over time, the "density" of interconnections has increased significantly [HUS09b]. This increase in density can be attributed to both multi-homing of subscribers as well as addition of new subscribers. With new subscribers, it implies that an S-BGP UPDATE message would need to be forwarded to an increasing number of peers over time. This requirement for a different signature per peer will potentially be difficult to manage with the current Internet growth pattern.

On a related note, there are processing and memory requirement implications with regards to transmission of RAs in BGP UPDATE messages. Studies have shown that RAs can increase the BGP UPDATE message size by as much as 800 percent [KEN03b]. In terms of memory, per peer requirement to store these RAs is about 30-35MB [KEN03b]. Again, with the increased density of ASes in the Internet today, this memory requirement may prove to be prohibitive.

In addition, an assumption relating to the environment, in which S-BGP must operate, creates logistical and operational challenges. The adoption of the protocol requires an upgrade to most of the Internet routers to enable them to support the computationally expensive protocol operations.

4.2.1 Concept of *Trusted* and *Non-trusted* ASes

As discussed earlier (see Section 1.2), there are two major types of ASes in the Internet, namely the "transit" ASes and the "stub" ASes. A transit AS mostly propagates BGP routes on behalf of other ASes, while a stub AS either originates or terminates BGP routes. An analogy for transit ASes would be major air traffic transit hubs like Chicago, New York, London, Toronto, Dubai, Los Angeles, etc. Similarly stub ASes could be analogous to smaller airports all over the world.

Like major air traffic hubs, these transit ASes are typically very large entities with strong financial backing and very strict security requirements as well as SLAs. They also carry a lot more of the Internet routing as well as data traffic and hence are critical to the operation of the Internet. If these ASes become compromised in some way or another, there is a very significant impact on the operation of the Internet. Typically transit ASes have very strict security operational policies and procedures in place so it is highly unlikely that they would intentionally participate in an attack on the Internet themselves. Considering their size, security requirements, self-interest in securing BGP and their financial capacity to undertake software and hardware upgrades, we treat these transit ASes as "special" ASes. For the rest of this analysis and the thesis, we call them *trusted* ASes. We may also use the terms *trusted* AS and transit AS interchangeably.

The stub ASes are typically customers of the transit ASes and are typically much smaller in size and have less security procedures. At the same time, there are many more of them as compared to transit ASes; 85% of the Internet today consists of stub ASes [HUS11d]. They are also spread out in many different geopolitical regions throughout the world. As a result of a variety of incentives (see Section 2.3), stub ASes are also more susceptible to being compromised and can use BGP to launch attacks against the Internet. However, it is also important to note that these ASes have far less incentives, and capacity (financial or logistics) and will power to adopt any security mechanisms for BGP that would put unnecessary stress on them, either in the shape of new financial requirements or operational procedures. Even if all of them were to agree to adopt a complex BGP security solution, logistically it would take many years for all stub ASes in the Internet to adopt and deploy the new standards. Therefore it is important that our security designs view these as primary constraints. For the rest of this analysis and the thesis, we call stub ASes as *non-trusted* ASes. We may also use the terms *non-trusted* AS and stub AS interchangeably.

4.2.2 Use of *Trusted* and *Non-trusted* ASes in C-BGP

S-BGP mandates that each AS should validate the AS-PATH and address prefix origination of each UPDATE message. This redundancy can be avoided if an AS can trust the verification done by previous ASes. For this purpose, we assume that a number of ASes are *trusted* by the neighboring ASes. Verification of attestations performed by these *trusted* ASes are reused by *non-trusted* ASes. Therefore, we propose modification to be made to S-BGP in order to include the following:

- A number of ASes are chosen to be *trusted* by neighboring ASes in the network. The *trusted* ASes have master certificates that distinguish them from the rest of the ASes.
- When a *trusted* AS receives an update, it performs verification of the full path and address prefix as specified by the S-BGP. Then, it signs the full path with its private key.
- When a *non-trusted* AS receives an update it checks only the AS-PATH portion between itself and the last *trusted* AS. Address prefix origination is verified only if no *trusted* AS is in the path of the update.

These modifications reduce the burden of the processing overhead required by S-BGP. Indeed, as shown in the next sections, the number of verifications is reduced significantly when a limited number of *trusted* ASes are present in the network. A *trusted* AS could be a major AS in a country or continental region. It’s more realistic to believe that large ASes have the resources to acquire expensive hardware that meet the processing requirements of S-BGP. The protocol should be flexible enough to allow original and enhanced updates to be carried by the same message. Indeed, if an AS chooses not to trust any AS in the network, it can implement the verification of the full path and address prefix.

4.2.3 Digital Signature Operations

When an AS receives a BGP advertisement, it appends the next hop (i.e., the next AS to which it will re-advertise this prefix) to the AS-PATH and signs the new AS-path along with all previous route attestations. This provides assurance of the integrity and authenticity of the path. Upon reception of an S-BGP update, an AS has to verify the AS-PATH and the authenticity of the AS claiming to originate the address prefix. The number of verification operations required depends on the number of ASes in the AS-PATH. For each AS in the path, the receiving entity has to fetch the AS certificate from a local cache, and proceed to the verification of the authenticity of the AS signature included in the update. In addition, the AS is required to verify the legitimacy of the address prefix by checking the "right of use" certificate that authorizes the originating AS to announce the IP address. Processing the additional certificates, and validating their authenticity, however, would require current-generation routers to devote a significant amount of horsepower to that task, and would impact their ability to handle traffic.

If the size of the AS-PATH is *ps,* then *ps* verifications are required to validate the BGP message. When an update traverses a path to reach the destination, the network would have performed number of signature verifications given in (4.1), where k is only an index of summation.

$$\sum_{k=2}^{ps} k \qquad = \frac{\mathrm{ps}*(\mathrm{ps}+1)}{2} - 1 \tag{4.1}$$

Among the *ps* verifications performed by an AS, *ps*-1 verifications are redundant since they have been already performed by the previous AS in the path. If we assume that a receiving AS trusts the immediate previous AS, then it has to verify only the signature of the previous AS to validate the BGP update message.

In the following example, we describe how the new verification scheme operates. In

Figure 4-1, we assume that ASes 4 and 6 are *trusted* ASes in the network and AS1 announces an address prefix A1.

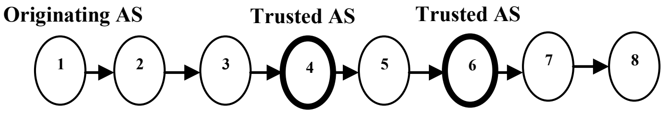

Figure 4-1: An example of BGP UPDATE message validation.

AS2, AS3, AS4 and AS6 will have to verify the full AS-PATH and AS1's address prefix (A1). They are required to perform 2, 3, 4 and 6 verifications, respectively. AS 5 has to verify only the signature of AS4 since AS 4 is *trusted* by AS5. This reduces the number of verifications performed by AS5 from 5 to 1. Similarly, AS7 and AS8 have to perform one and two verifications, respectively. The total number of verifications done by the network along this path is 19 instead of 35 in the original version of S-BGP.

4.3 Analytical Model

We develop and present an analytical model to analyze the impact of trust on the number of signature verification operations in the presence of a (random) percentage of *trusted* ASes in the network. This analysis is used to compute theoretical approximation of the performance metrics of the proposed alternative protocol and is important to prove that the proposed protocol significantly reduces the computational overhead associated with signature verification operations in S-BGP. The analytical model is also verified with a self-developed BGP simulator.

4.3.1 AS Topology

We simplify our analysis by modeling each autonomous system as a single node in the topology. In practice, most ASes encompass dozens or even hundreds of border and internal routers.

In order to make our analysis tractable, we chose to model the topology of the network as a set of ASes connected to each other with a constant number of links denoted as $avgL$. We model our network from the view of the origin of the update O as shown in Figure 4-2. We will use this topology model as the basis for the development of the analytical model.

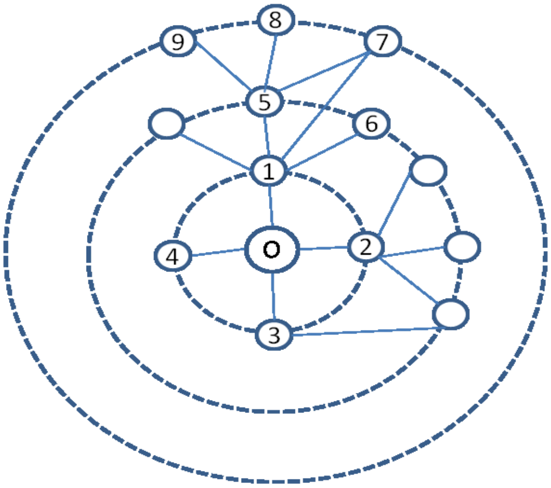

Figure 4-2: Network topology from the perspective of the origin of the UPDATE message.

We define α_n as the set of ASes that are n hops away from the origin of the update and therefore are required to perform n signature verifications to verify the update. We note that an AS can belong to one or more sets depending on how it's connected to the rest of the ASes. For example, AS 7 belongs to α_2 and α_3 since it's connected by a two hop path (O-1-7) and a three hop path (O-1-5-7).

4.3.2 Average Number of Updates Per AS and IP Prefix

For every received update, an AS processes the update and compares the path length to the path length stored in the local Routing Information Base (RIB) for that IP prefix. As noted earlier, AS-path length serves as the only metric for route preference.

Assuming there are no losses in the AS, we can write:

$$\sum incoming\ updates = \sum outgoing\ updates \tag{4.2}$$

Therefore,

$$\frac{1}{N}\sum incoming\ updates = \frac{1}{N}\sum outgoing\ updates \tag{4.3}$$

The previous equality can be written as:

$$avg\_updates = avg\_incoming\_updates = avg\_outgoing\_updates \tag{4.4}$$

For the remaining analysis, we will estimate the average number of outgoing updates. In our reference network topology (Figure 4-2), we assume that every AS has an average of *avgL* peers.

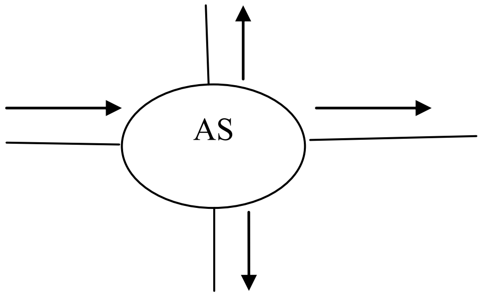

Figure 4-3: The n^{th} incoming update generates *avgL*-1 outgoing updates with probability $\frac{1}{n}$

For each incoming update from a peer, the AS generates *avgL*-1 updates if the received update is announcing the best route to the source IP prefix. Therefore, the first update generates *avgL*-1 updates (no update is sent on the link that is learnt on). The second update for the same IP prefix has 50% probability to be better (shorter) than the previous update, and therefore, will generate $\frac{1}{2} * (avgL - 1)$. Similarly, a subsequent update n will generate *avgL-1* with a probability$\frac{1}{n}$.

Assuming that an incoming update for an IP prefix is received at most once per link, we find the average number of outgoing, (and incoming), updates per AS for an IP prefix:

$$avgU = (avgL - 1) + \frac{1}{2}(avgL - 1) + \frac{1}{3}(avgL - 1) + \cdots . + \frac{1}{(avgL)}(avgL - 1) \qquad (4.5)$$

This can be simplified to the equation in (4.6), where *k* is an index of summation.

$$avgU = (avgL - 1) \sum_{k=1}^{avgL} \frac{1}{k} \qquad (4.6)$$

4.3.3 Number of ASes That are n Hops Away from Origin

In this section, we estimate the dimension of the set α_n.

Since the origin is connected to *avgL* peer ASes, we can write $|\alpha_1| = avgL$.

For $n \geq 2$, we have:

$$|\alpha_n| = (avgL - 1) * |\alpha_{n-1}| * prob(AS\ verifying\ update) \qquad (4.7)$$

This can be explained by the fact that each AS belonging to the set α_{n-1}will propagate the update to the remaining$(avgL-1)*|\alpha_{n-1}|$ neighbors; however, the update is verified if and only if it's carrying a new route or a better route.

Upon receiving an update, an AS either verifies the update because it's carrying a new route or best update or drops the update because it's not carrying a better (shorter) route. Therefore, we can formalize this obvious observation as:

$$1 = prob(AS\ verifying\ update) + prob(AS\ not\ verifying\ update) \tag{4.8}$$

Furthermore, an AS receiving an update doesn't verify it if the following two conditions are met:

1. The receiving AS belongs to a set α_i with index below or equal to *(n-1);*
2. The received update is not carrying a new or best route.

If both conditions are met, we can state that the receiving AS has already received and processed an update for that IP prefix.

Furthermore, an AS is belonging to set α_{n-1} if it has a neighbor (one of *avgL*-1 neighbors) that is in the set $\bigcup_{i\le n-2}\alpha_i$. Therefore:

$$prob.(AS\ not\ verifying\ update) = prob(not\ best\ update)*(avgL-1)*\frac{\sum_{i=1}^{n-2}|\alpha_i|}{N} \tag{4.9}$$

Assuming that there is one best update received among *avgU* updates received by an AS, we can simply estimate $probability(not\ best\ update)$ as:

$$prob(not\ best\ update) = \frac{avgU-1}{avgU} \tag{4.10}$$

From the previous equations, we can express α_n as:

$$\alpha_n = (avgL - 1) * \alpha_{n-1} * \left(1 - \frac{avgU-1}{avgU} * (avgL-1) * \frac{\sum_{i=1}^{n-2}|\alpha_i|}{N}\right);\ \mathrm{n} \geq 2 \quad (4.11)$$

We note that if $\left(1 - \frac{avgU-1}{avgU} * (avgL-1) * \frac{\sum_{i=1}^{n-2}|\alpha_i|}{N}\right) < 0$, then,

$$\alpha_n = 0 \quad (4.12)$$

4.3.4 Average Number of Signature Verifications

In this section, we develop a theoretical formula to calculate the average number of signature verifications performed by each AS in the presence of a random percentage of *trusted* ASes in the network. Recall that in our proposal, a *non-trusted* AS will verify the path up to the last *trusted* AS in the AS-PATH.

We denote the ratio of *trusted* ASes with variable *x*.

Once a *non-trusted* AS *NT_AS*$_n$ receives an update from *AS*$_{n-1}$, it does check the signature of *AS*$_{n-1}$, and repeats the same verifications performed by *AS*$_{n-1}$ in the case the latter is not *trusted*, which will occur with a probability (*1-x*).

Therefore, if *NT_SigV*$_n$ denotes the number of verifications performed by *non-trusted* AS *NT_AS*$_n$, we can write the following relationship between the average number of signature verifications performed by *NT_AS*$_n$ and *AS*$_{n-1}$:

$$NT\_SigV_n = (1-x) * NT\_SigV_{n-1} + 1 \quad (4.13)$$

with $NT\_SigV_0 = 0$

NT_SigV_n is an arithmetic-geometric sequence that the general term can be expressed as:

$$NT\_SigV_n = \frac{1}{1-(1-x)} + (1-x)^n * (0 - \frac{1}{1-(1-x)}) \quad (4.14)$$

By simplifying the term NT_SigV_n, we find:

$$NT\_SigV_n = \frac{1-(1-x)^n}{x} \tag{4.15}$$

On the other hand, a *trusted* AS *T_ASₙ* performs the full path verification, resulting in *n* signature verifications. Hence, we can write the number of signature verifications performed by *trusted* ASes as:

$$T\_SigV_n = n \tag{4.16}$$

Since the network has *x trusted* ASes and *(1-x) non-trusted* ASes, we can write the number of signature verifications performed by an AS at distance *n* of the origin as:

$$SigV_n = x * T\_SigV_n \; + \; (1-x) * NT\_SigV_n \tag{4.17}$$

This can be simplified as:

$$SigV_n = n * x + \; (1-x) * \frac{1-(1-x)^n}{x} \tag{4.18}$$

Therefore, the average number of signature verifications performed by each AS per IP prefix is:

$$avgSigVerificationsPerPrefix = \sum_{n=1} \frac{|\alpha_n| * SigV_n}{N} \tag{4.19}$$

With α_k is the number of ASes that receives the update with an AS-PATH length equal to *k*. These factors are estimated in the previous section.

Assuming that each AS advertises one IP prefix, we can write the general formula for the average signature verifications performed by each AS in the presence of a percentage x of *trusted* ASes:

$$avgSigVerifications = \sum_{n=1} |\alpha_n| * [n * x + \; (1-x) * \frac{1-(1-x)^n}{x}] \tag{4.20}$$

4.3.5 Average Number of IP Prefix Validations

In C-BGP, a *non-trusted* AS performs an IP prefix validation if there is no *trusted* AS in the AS-PATH. Therefore, the number of IP prefix validations performed by a *non-trusted* AS NT_AS_n can be expressed as:

$$NT\_prefixV_n = (1 - x)^n \tag{4.21}$$

Similarly, a *trusted* AS always verifies the IP prefix. Therefore, we can write the number of IP prefix validations performed by *trusted* ASes as:

$$T\_prefixV_n = 1 \tag{4.22}$$

Therefore, the number of IP prefix verifications performed by an AS far by *n* hops from the source of the update can be expressed as:

$$prefixV_n = x * T\_prefixV_n + (1 - x) * NT\_prefixV_n \tag{4.23}$$

$$prefixV_n = x + (1 - x)^{n+1} \tag{4.24}$$

The average numbers of IP prefix validations performed by each AS in the presence of a percentage *x* of *trusted* ASes can be calculated as:

$$avgPrefixValidations = \sum_{n=1} |\alpha_n| * [x + (1 - x)^{n+1}] \tag{4.25}$$

4.3.6 Average Number of Public Keys Required

In our model, each AS is required to store the certificates/public keys of its neighbors and the public keys of second level neighbors if the first level neighbor that connects it to the second level neighbor is not *trusted* and so on.

Therefore, we can write:

$$avgKeys = (avgL) + (1-x) * (avgL) * (avgL-1) + (1-x)^2 * (avgL) * (avgL-1)^2 + \cdots$$

Assuming that there are *z* levels that constitute the network, the expression can be written as:

$$avgKeys = (avgL) * \sum_{k=1}^{z}((1-x) * (avgL-1))^{k-1} \tag{4.26}$$

And then the expression can be simplified to:

$$avgKeys = (avgL) * \frac{(1-((1-x)*(avgL-1))^{z})}{1-(1-x)*(avgL-1)} \tag{4.27}$$

Knowing that avgKeys=N for *x*=0, we can calculate the number of levels *z* from the formula:

$$N = (avgL) * \frac{(1-(avgL-1)^{z})}{1-(avgL-1)} \tag{4.28}$$

Therefore,

$$z = \frac{\ln(1-\frac{N}{avgL}*(1-(avgL-1))}{\ln(avgL-1)} \tag{4.29}$$

For N=2000, the value of *z* is: 6.7.

4.4 BGP Simulator

We developed a discrete event simulation model which can simulate propagation of BGP UPDATE messages and produce metrics results which can be compared with the analytical model above. In terms of requirements, we wanted the simulation model to accomplish the following goals:

1. Simulation package must produce results which are statistically similar to already published results by independent authors who have contributed in the field of BGP security. This goal formed the bedrock of the simulation analysis.

2. The simulation must focus exclusively on EBGP UPDATE messages and not have IBGP UPDATE messages influence the propagation of EBGP UPDATE messages.
3. It must model each AS as a node in the system.
4. It must allow for a random topology with the option of the user to specify average degree of connectivity for each AS.
5. It must allow us to calculate metrics while processing BGP UPDATE messages.
6. It must allow for different models for propagation of BGP UPDATE messages in the network. The modeling of propagation of BGP UPDATE messages is important as it has direct impact on the number of signature verifications performed by the overall network. For example, if the simulation model applies sequential processing of updates, the resulting UPDATE messages will result in shortest paths. On the other hand, random propagation of UPDATE messages will result in a more realistic scenario.
7. It must model propagation of a BGP UPDATE from an AS who hijacks an IP prefix. It must also calculate the "damage" caused by such an UPDATE. This is defined as the the ratio of ASes that get their RIB polluted by the malicious BGP UPDATE message.
8. It must also model sending packets destined to the hijacked IP prefix and then determine the ratio of ASes that get their traffic redirected towards the malicious AS, which measures the success of the attack.

To help expedite research and to have trustable results, we did consider using existing BGP simulation packages. However, we did not find sufficient flexibility in the packages to accommodate several changes that were required to meet the goals. It was therefore decided that a new simulator be designed and implemented. In order to gain trust in the system, it was paramount to achieve goal number 1 as first priority of the simulation system.

The design philosophy behind the new simulation package was to keep it simple, modular, scalable and easily extendible. The core modules of the simulator are as follows:

- Topology management module
- Update table and RIB table
- Network-wide UPDATE message propagation module
- Signature validation and metrics management module
- Path selection algorithm
- Generation of spoofed UPDATE messages

Each of these modules is independent and can easily be enhanced or replaced by different implementations on as needed basis. We present the details of each module next. We will then also present and discuss a flow-chart figure of the simulation package.

4.4.1 Topology Management Module

This module is responsible for generation of the AS topology to be used by the simulation system. It creates a topology with the user specified number of ASes. In terms of interconnections between the ASes, the module allows for two modes, *random topology mode* and a *pre-defined topology mode*. In *random topology mode*, the module creates user specified number of connections/degrees, *d*, between each AS. It uses a uniform distribution to select ASes (with less than *d* connections each) that can be connected to each other. The module stops once each AS is connected with *d* other ASes. The random mode is useful for comparison with the analytical model presented in this thesis. In *pre-defined topology mode,* the module imports a text file with a table with predefined connectivity between ASes. This file can be created offline with a predefined AS relationship. The pre-defined topology mode is useful for simulating

Internet topology as well as to compare with existing research results in the area of BGP security.

4.4.2 Update Table and RIB Table

The Update table is a global entity and it maintains a static snapshot of all the BGP UPDATE "messages" that are needed to be processed by the ASes in the system. Each of the UPDATE "messages" carries the following information:

- The IP prefix being announced.
- Sender AS of the UPDATE message,
- Intended recipient AS of the UPDATE, and
- AS-PATH (i.e. ASes traversed by the update).

In our model, each AS only advertises a single IP prefix to the rest of the network. When the simulation system starts up, each AS announces its prefix to its directly connected peers by adding UPDATE "messages" to the Update Table, one for each peer. Similarly, when an AS processes an UPDATE message and needs to forward the message to its peers, it adds the update version of the UPDATE "message" to the Update Table. The simulation terminates when the UPDATE table processing is completed.

The RIB table is maintained on a per AS basis and is used to maintain the best route from each AS to all other ASes in the network. The RIB table gets populated as UPDATE messages are processed. If the incoming UPDATE message contains a better route (we discuss path selection algorithm later) than the locally stored route in the RIB, RIB is updated with the route in the incoming UPDATE message. Else, the incoming message is discarded. It is also discarded if the signature verification fails.

4.4.3 Network-wide UPDATE Message Propagation Module

If the RIB is updated with the route in the incoming UPDATE message, the new route needs to be propagated to the directly connected peers. This is achieved by adding an UPDATE message to the UPDATE Table. It is very important to model the nature of the timing involved in the propagation of the UPDATE message. Consider a simple example in Figure 4-4 below.

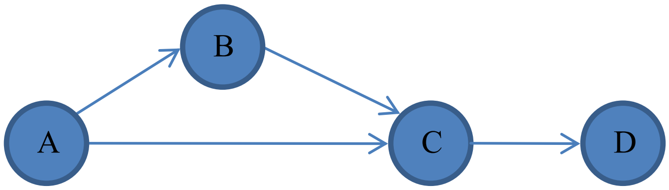

Figure 4-4: An example of topology to model propagation of BGP messages.

There are two routes to go from A to D, namely ACD and ABCD. The selection algorithm is based on shortest path. In this case, route ACD is better than ABCD. Since this is discrete event simulation, the simulator must choose and process signature verifications for either route AC first or route ABC first. If the simulator is designed in such a way that it always processes route ABC first versus AC, then AS D will need to receive and process both routes ABCD and ACD. This will require more signature verifications. On the other hand, if the simulator always chooses route AC first, then AS D will only receive route ACD and there will be fewer signature verifications needed. Without prior knowledge of the entire routing table, it is not possible to capture the propagation of route UPDATES.

In our model, we solve this problem by applying uniform distribution to randomly select which UPDATE to process from the UPDATE table. This ensures that the AS graph is not traversed in a predetermined fashion. In the example above, route AC may be processed ahead of route AB or route BC may get processed ahead of route AC, etc.

4.4.4 Signature Validation and Metrics Management

In each UPDATE message in the Update Table, the simulator stores the number of signatures that need to be verified by the AS receiving this UPDATE. When the UPDATE message is added to the Update table, this count is set to 2, one for the IP prefix validation and another one for AS signature. As each AS propagates this UPDATE message, this count gets incremented. The exact algorithm depends on whether there is any *trusted* AS in the AS-PATH or not. For each UPDATE message, following metrics are computed:

- Number of IP prefix validations conducted so far.
- Number of AS-PATH verifications.
- Number of keys needed.

4.4.5 Path Selection Algorithm

The route selection algorithm is based on shortest AS-path and only the best route is propagated by each AS in the network. The decision to use shortest AS-path algorithm is justified by the fact that there is no comprehensive database that gives a complete view of the local preferences of each AS. It's hard if not impossible to learn all the local policies of autonomous systems on the Internet because local policies are kept confidential as they reflect commercial relationships between service providers and their customers.

The choice of shortest AS-path is also consistent with several existing studies about BGP security and simulation models. It is important to highlight that the analysis does not assume any

AS-PATH prepending or other similar techniques employed sometimes by AS administrators to influence the route selection algorithm. For the purposes of our analysis, the technique of AS prepending does not influence the metrics as the signature verification is still only performed once for each AS in the AS-PATH.

4.4.6 Generation of Spoofed UPDATE Messages

Spoofing of UPDATE messages is achieved by changing the source AS for a randomly selected UPDATE message to a value different from the true origin AS. This introduces the concept of prefix hijacking in the system. When an UPDATE with a spoofed IP address is processed in step 5 above, it is discarded as it fails valid signature check. However, if this malicious UPDATE is being received from a *trusted* AS (e.g. a *trusted* AS has been compromised), the UPDATE will be accepted and the attack considered as successful. There will be calculations performed to determine the "damage" done by this spoofed UPDATE.

4.4.7 Flowchart for the Simulation Model for BGP Route Distribution

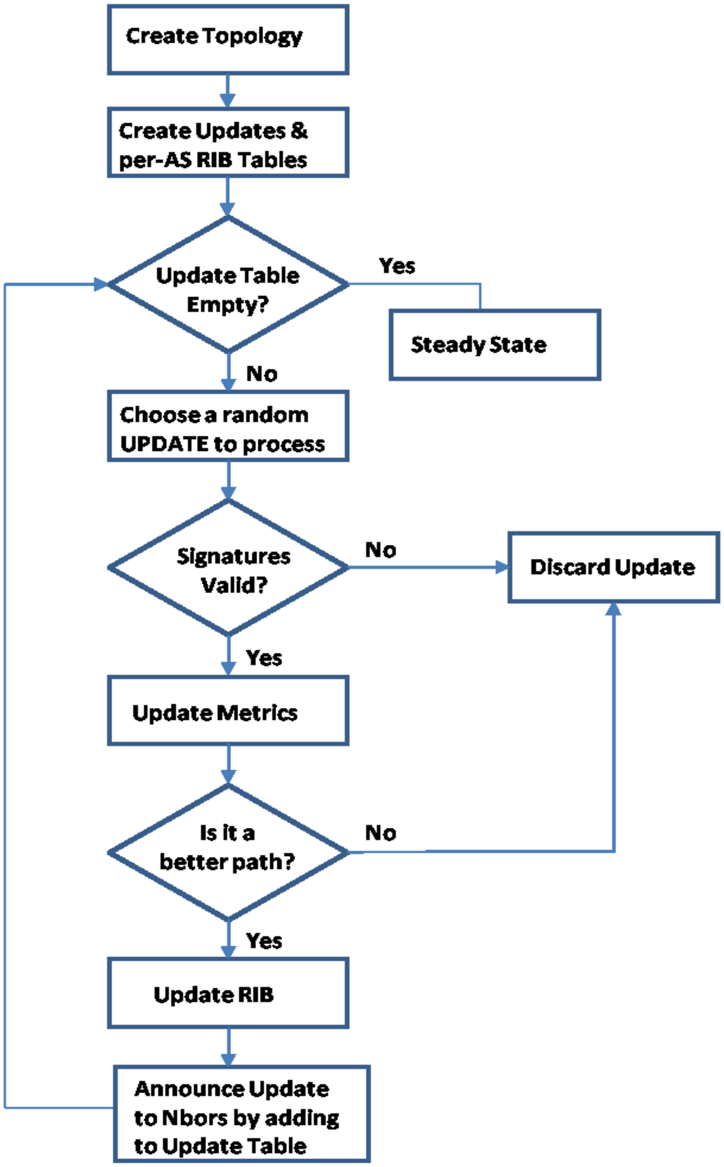

Figure 4-5 depicts the flowchart diagram for the simulation model. The sequence is as follows:

1. The topology module is initialized as per user requirements and the desired topology is created in the database.
2. The list of ASes considered as "*trusted*" is also generated.

3. Each AS generates an UPDATE message based on the IP prefix address it wishes to announce and adds the message as an announcement in the Update Table for each of its peers. At this point, the Update Table is populated with, on average, *d* UPDATE messages for each AS in the topology. Also, RIB is initialized with no route information at startup.
4. When/if the Update table is empty, the simulation quits.
5. As long as Update Table is not empty, the network-wide update propagation mechanism kicks in and randomly selects an UPDATE message from the Update Table to process.
6. The signatures embedded in the UPDATE message are verified. In the case of a malicious UPDATE (in the case of an attack), this verification will fail and the UPDATE is dropped.
7. If the signatures are valid, the UPDATE is processed and the desired metrics updated accordingly.
8. The selection algorithm kicks in and determines whether the UPDATE message contains a route that is better than the existing route in the RIB. In case, the route in the UPDATE message is longer than the route in RIB, the UPDATE message is discarded.
9. If the route in the UPDATE message is better than the route in RIB, RIB is updated with the new route.
10. Since there has been a change in the route to the destination, the new route must be advertised to the neighbors. This is accomplished by adding a new UPDATE for each peer to the Update Table. The simulation continues back at 3.

The route distribution ends when the network reaches the steady state, that is, each AS has learnt the best route to reach every other AS in the network.

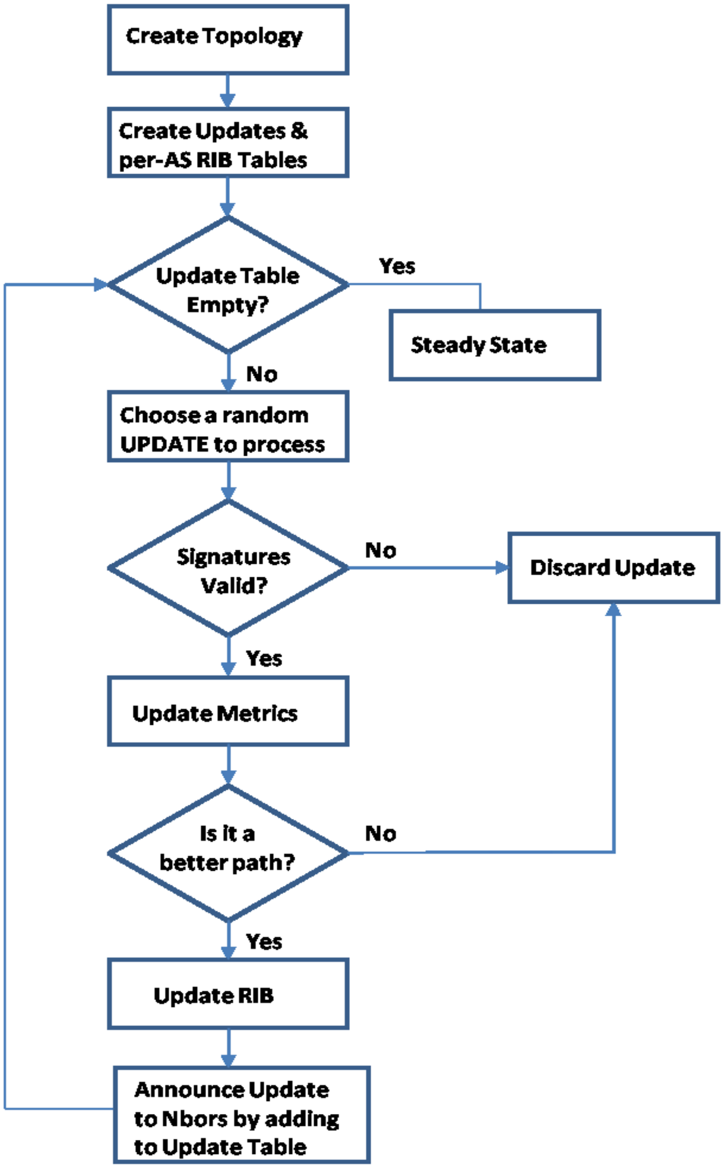

Figure 4-5: Simulation flowchart.

4.5 Simulation Results

In this section, we present data collected from the simulation model as well as from the analytical model.

4.5.1 AS-PATH Verifications

In Figure 4-6, results demonstrate that on average 20% of *trusted* ASes in the network can reduce the number of AS-PATH signature verifications by approximately 33%. In the experimental results, average number of signature verifications is approximately 18,000 with 0 *trusted* ASes. This number decreases to approximately 12,000 when 20% of the ASes are made *trusted*. Both the analytical and simulation graphs follow the same pattern which serves as a strong validation of the analytical model developed earlier. It is interesting to note that beyond a certain ratio value (~35%), the average number of signature verifications starts to increase again. This phenomenon can be explained by the fact that as more *trusted* ASes are introduced in the network, the number of full signature verifications increases. Basically, the average number of signature verifications is the same at ratio values of 0% and 100%, as in both cases all ASes participate in full AS-PATH verifications.

Although both analytical and experimental results follow the same pattern and are relatively in agreement with each other, there are more signature verifications done in analytical model as compared to simulation model. This is because in analytical model, calculation of average number of updates, *avgU*, assumes that the update for the same prefix is received from all interfaces and neighbors. In simulation, the updates are only accepted from < avgL neighbors. This overestimation of avgU results in difference between analytical and simulation results.

In Figure 4-7, we display average number of AS-PATH signature verifications required by the *non-trusted* ASes in the network. As predicted, this number drops significantly as the number of *trusted* ASes is increased in the network. If we consider the case of 20% of the ASes being *trusted*, the average number of signature verifications required by *non-trusted* ASes drops from approximately 18,000 to 10,500, which is a 42% reduction in number of verifications. These results are very encouraging for the proposed scheme and are attractive for many small ASes who may not be willing or capable to perform full signature verifications.

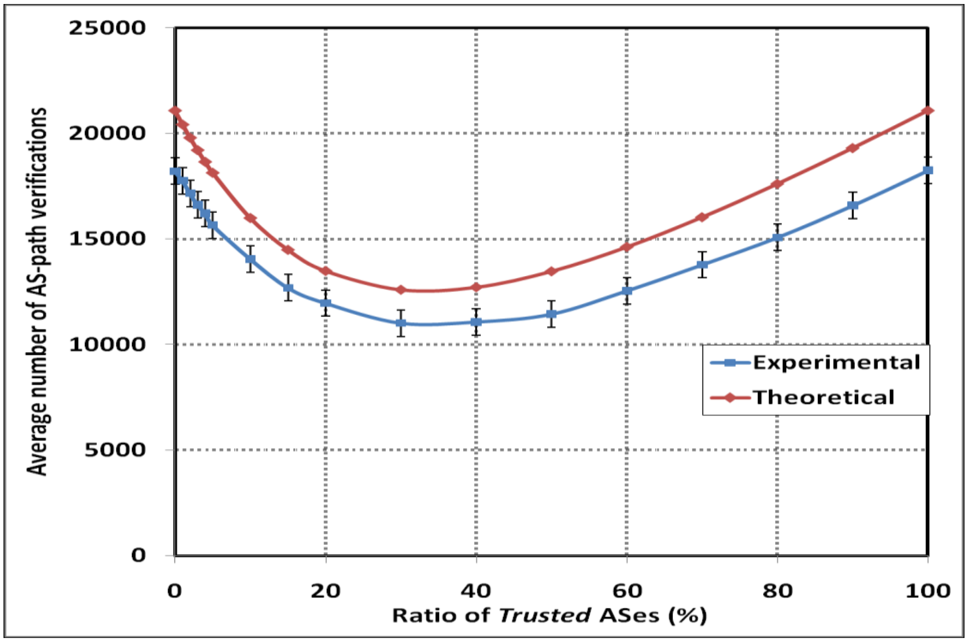

Figure 4-6: Average number of AS path verifications performed by proposed scheme.

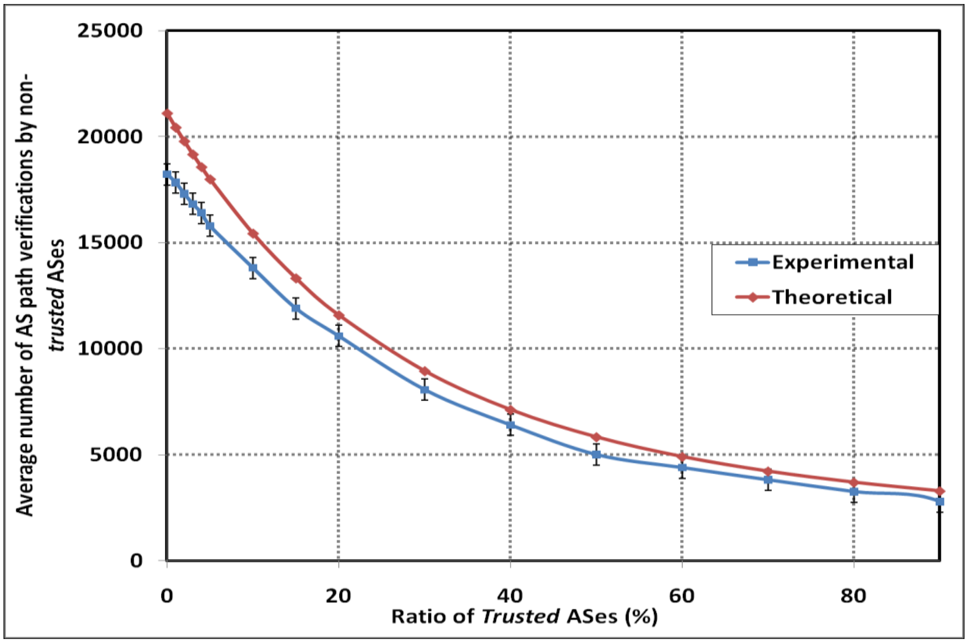

Figure 4-7: Average number of AS path verifications performed by *non-trusted* ASes.

4.5.2 IP Prefix Validations

As shown in Figure 4-8, the average number of IP prefix validations is reduced by approximately 60% when 20% of the ASes are *trusted*. This can be explained by the fact that only *trusted* ASes in the AS-PATH are required to perform IP prefix verifications. All *non-trusted* ASes in the path are not required to perform IP prefix validation. Also, when the number of *trusted* ASes is increased beyond approximately 30%, the number of IP prefix validations starts to increase. This is because the increased number of *trusted* ASes means more IP prefix validations in the network. Also, the results from analytical model and experimental model are in close agreement with each other.

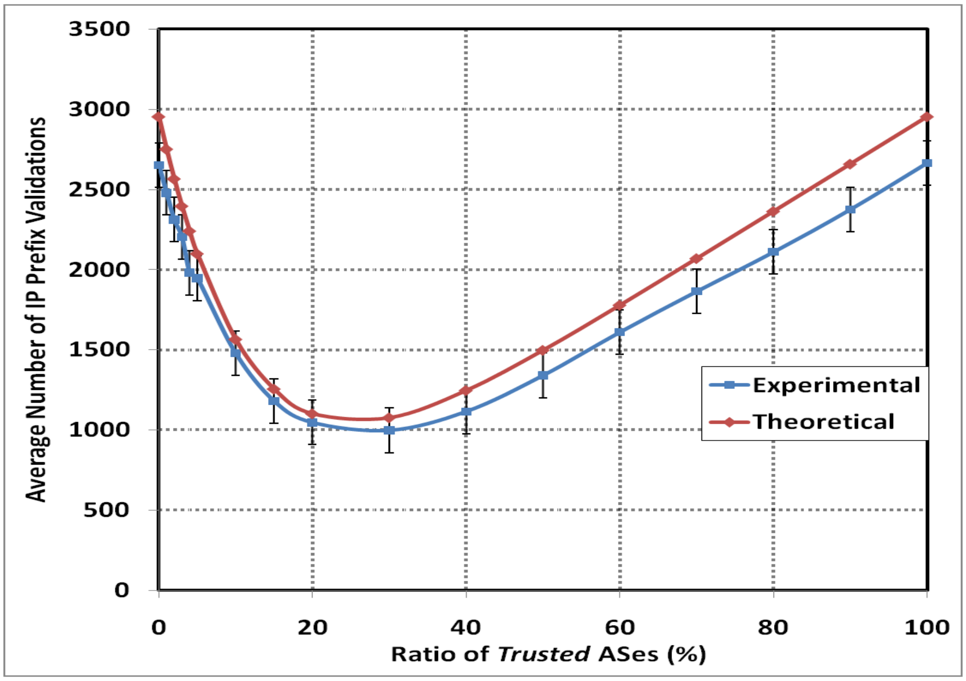

Figure 4-8: Average Number of IP prefix verifications performed by all ASes.

As expected, the average number of IP prefix validations performed by *non-trusted* ASes reduces dramatically as more *trusted* ASes are introduced in the network. With approximately 60% of the ASes in the network being *trusted*, there are virtually no IP prefix validations being performed in the network by *non-trusted* AS. It also means that with 60% of ASes *trusted*, almost every AS-PATH has at least one *trusted* AS in it who is performing the IP prefix validation for this AS-PATH.

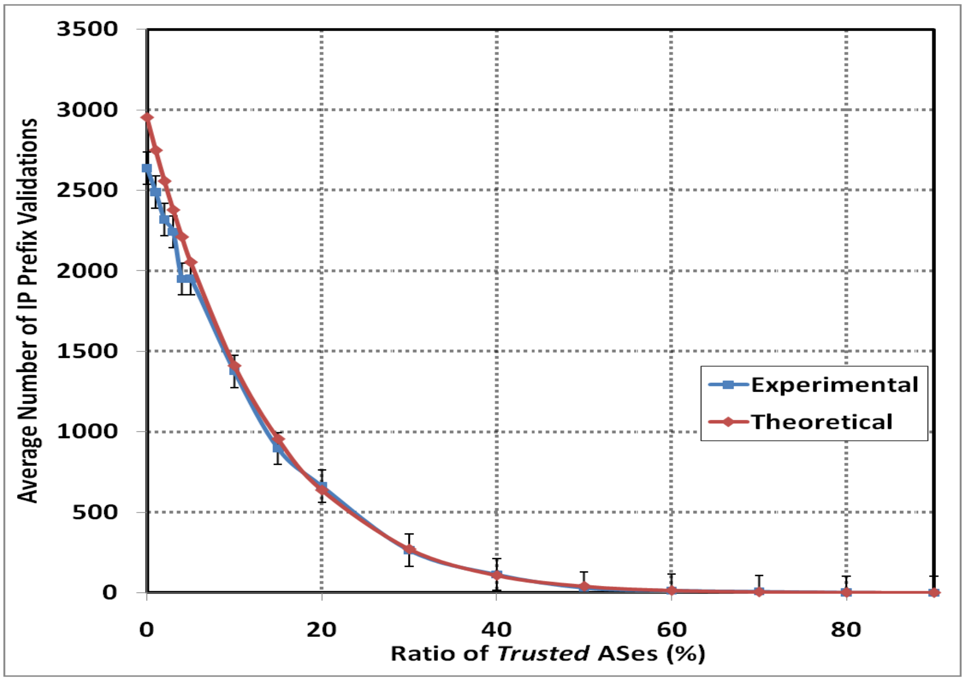

Figure 4-9: Average Number of IP Prefix Validations by *non-trusted* ASes.

4.5.3 Average Number of Public Keys

Using the proposed scheme, the average number of public keys required to be known by each AS is reduced significantly. As shown in Figure 4-10, the average number of public keys is reduced by approximately 60% when 20% of the ASes are *trusted*. With less average number of public keys required by the system, it alleviates the key storage, key management and memory requirements on individual ASes. Again, when the number of *trusted* ASes is increased beyond approximately 30%, the number of public keys needed starts to increase. This is because the increased number of *trusted* ASes means more keys needed to perform validations in the network. Again, both theoretical and experimental results are in close agreement with each other.

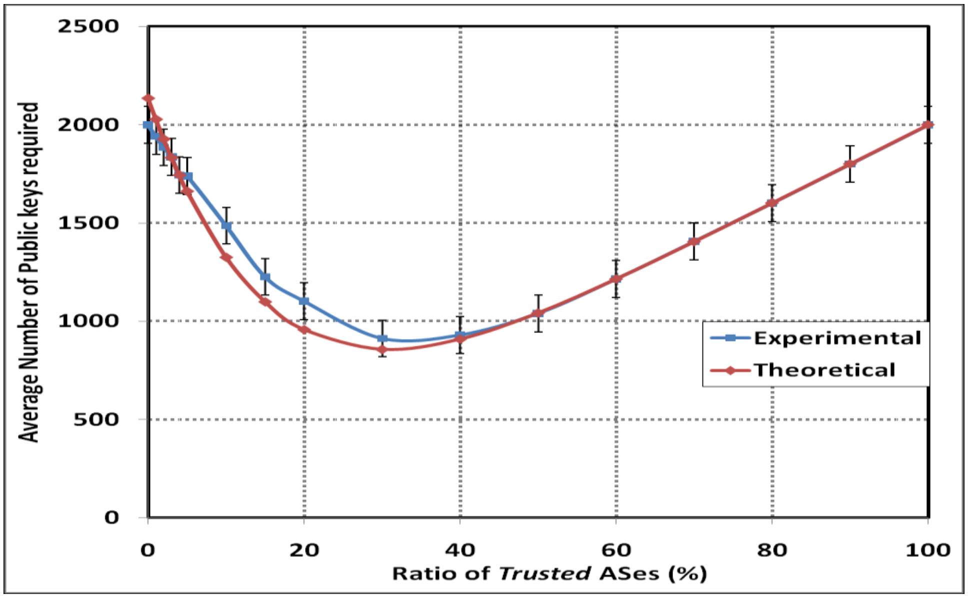

Figure 4-10: Average Number of Public Keys required in the system.

As expected, the average number of public keys needed by *non-trusted* ASes reduces dramatically as more *trusted* ASes are introduced in the network. From Figure 4-11, it is evident that with approximately 60% of the ASes in the network being *trusted*, each *non-trusted* AS needs to know of very few public keys.

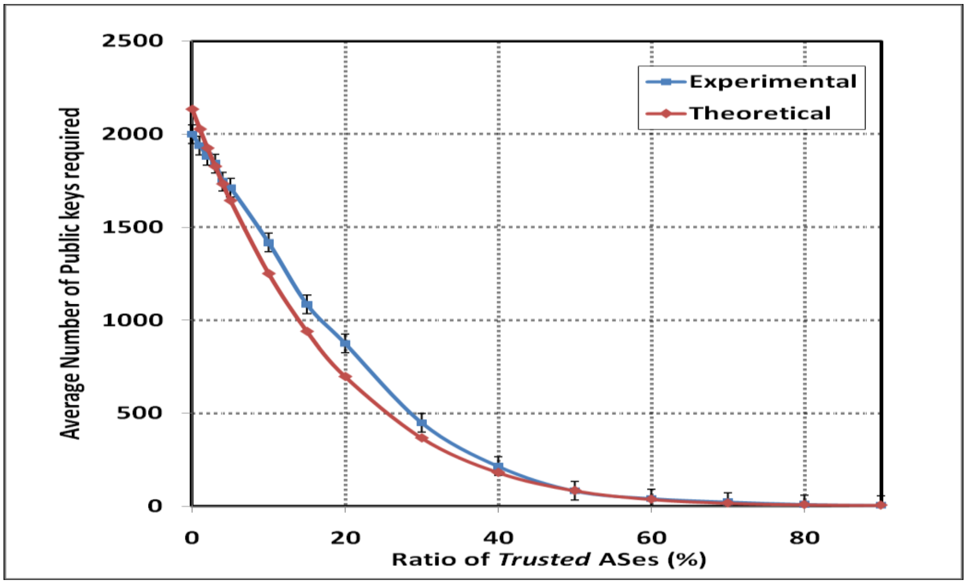

Figure 4-11: Average number of public keys required by *non-trusted* ASes.

4.6 Conclusion

In this chapter, we presented an evaluation of a new BGP update validation scheme that reduces the processing load of the routers deploying S-BGP like protocol. We developed and presented an original analytical model to analyze the impact of trust on the number of signature verification operations in the presence of a (random) percentage of *trusted* ASes in the network. The model was used to determine number of signature verifications, number of IP prefix validations and average number of public keys required in the BGP network. We verified results produced from the analytical model closely match those produced by running simulations on a self-developed simulator. The study showed that when 20% of *trusted* ASes are present in the network, the number of AS-PATH verifications is reduced by almost 33%, and the average number of IP prefix validations is reduced by 60%. For the same percentage of trusted ASes, the average number of public keys is also reduced by 60%.

Chapter 5

Security Analysis of C-BGP

5.1 Introduction

As discussed in Chapter 4, C-BGP defines a layer of ASes that are chosen to be globally *trusted* by all other ASes in the network. The *trusted* ASes have master certificates that distinguish them from the rest of the ASes. When a *trusted* AS receives an update, it performs full verification of the path and address prefix as specified by S-BGP. Then, it signs the full path with its private key. When a *non-trusted* AS receives an update it checks only the AS-PATH portion between itself and the last *trusted* AS.

These modifications reduce the burden of the processing overhead required by S-BGP. As discussed in Chapter 4, there are significant savings in terms of number of signature verifications as they are reduced significantly when a limited number of *trusted* ASes are present in the network. This scheme works very well as long as the *trusted* ASes are indeed always trustworthy. Due to the security risks and economic benefits involved, the notion of *trusted* ASes may not be readily accepted by other ASes in the Internet. This is because from an operational point of view, there is no way to guarantee that a *trusted* AS itself cannot be malicious or compromised. However, if security impact of a malicious *trusted* AS can be studied, then there is objective data to assist in decision making of the various ASes. In this chapter, we aim to study the security impacts of a malicious trusted AS and answer the following questions:

1. On average, how much control plane damage can be caused if one of the *trusted* ASes in the network becomes malicious? This can be expressed as the ratio of ASes which get their RIBs polluted by the malicious BGP announcement, as compared to rest of the network.
2. On average, how much data plane damage can be caused if one of the *trusted* ASes in the network becomes malicious? This can be expressed as the ratio of ASes that get their traffic redirected towards the malicious AS.
3. Is there a minimum percentage ratio of ASes which can be made *trusted* and network relatively immune to an attack by a malicious *trusted* AS? At this ratio value of *trusted* ASes, what is the percentage savings in terms of number of signature verifications as compared to a network where all nodes perform IP address check as well as full path verifications.
4. How sensitive is the damage to the number of *trusted* ASes becoming malicious? For example, how does the security degrade when more than one *trusted* AS in the network become malicious?

For the above analysis, we assume following scenario: A *trusted* AS gets compromised by an attacker who converts it into a malicious AS by advertising an IP prefix that belongs to a different AS, denoted as the victim AS. We develop an original analytical model to measure two metrics: 1) the ratio of ASes which get their RIB polluted by the malicious BGP announcement and, 2) the ratio of ASes that get their traffic redirected towards the malicious AS, which measures the success of the attack. We also enhance the simulation model to conduct an analysis on our topology and provide experimental results. We also provide a commentary on the similarities and differences between the results from analytical model and the simulation model.

Recall that in C-BGP protocol, a *trusted* AS will perform full verification of the BGP UPDATE message and therefore will be able to detect any spoofed UPDATE messages. Therefore, success of IP prefix hijacking attacks depends on the ratio of *trusted* ASes versus *non-trusted* ASes in the network.

5.2 Control-plane "Damage" Performed by a Malicious AS

We define control plane damage in terms of ratio of ASes who get their RIB databases corrupted by malicious BGP announcements. In this section, we present the analytical model for this ratio. We shall present analysis for data plane damage in Section 5.3.

5.2.1 Analytical Model

Let's use *m* to denote a malicious AS who hijacks an IP prefix of a victim AS *v* and advertises it to the rest of ASes in the network.

We define $c(m, v, a)$ as follows:

$c(m, v, a) = 1,$ if RIB of AS a gets corrupted after m hijacks the IP prefix of v

$c(m, v, a) = 0,$ otherwise

We also define $C(m)$ as the average ratio of ASes with corrupted RIBs after a malicious (*trusted*) AS *m* hijacks the IP prefixes of all the other ASes in the network. N denotes the total number of ASes in the network.

$$C(m) = \frac{\sum_v \sum_a c(m,v,a)}{N} \tag{5.1}$$

In order to estimate the value of $C(m)$, we calculate the number of ASes that receive UPDATE messages containing the hijacked IP prefix being advertised by the malicious AS. This UPDATE message must not have any *trusted* AS in the AS-PATH, else the UPDATE message would be dropped by the *trusted* AS in the AS-PATH. This quantity, $C(m)$, is similar to the average number of keys calculated earlier (see Chapter 4, Section 4.3.6). It can be written as:

$$C(m) = \frac{1}{2*N} * (avgL) * \frac{(1-((1-x)*(avgL-1))^{z})}{1-(1-x)*(avgL-1)} \tag{5.2}$$

We used the coefficient $\frac{1}{2}$ to take into consideration the probability that the hijacked IP prefix UPDATE succeeds only 50% of the time. Recall that if the AS-PATH from AS *v* to AS *a* is shorter than the AS-PATH from AS *m* to AS *a*, then AS *a* will not populate its RIB with the UPDATE message being received from AS *m*. In a perfectly random distribution of connectivity between ASes and random locations of ASes *m*, *v* and *a*, there is an equal probability that AS-PATH from AS *v* to AS *a* is shorter or longer than the AS-PATH from AS *m* to AS *a*.

For the remaining analyses, c denotes the average ratio of ASes with a polluted RIB.

5.2.2 Simulation Model and Results for Control-plane Damage

We extended our simulation model to incorporate flooding of spoofed UPDATE messages via a randomly selected malicious AS. Figure 5-1 depicts the flowchart diagram for spoofed UPDATEs and calculations pertaining to resulting damage (in terms of ASes with polluted RIBs) from the spoofed UPDATES. The sequence is as follows:

1. We let the simulation execute and the entire routing table reach steady state.

2. A *trusted* AS is randomly selected as compromised and becomes malicious. We use uniform distribution model so that each AS has equal probability of being selected malicious.
3. The AS hijacks a random IP prefix belonging to some other AS and starts advertising it in the network as if it owns it.
4. The IP prefix is propagated in the network and each AS uses the Path Selection Algorithm to accept or reject the spoofed UPDATE message.
5. For each AS, we inspect its RIB to determine whether or not it accepted the spoofed UPDATE.
6. The ratio of ASes which accepted the spoofed UPDATE is calculated.

Steps 1-6 above are repeated for all *trusted* ASes, and the ratio in step 6 is averaged over the total number of *trusted* ASes in the network.

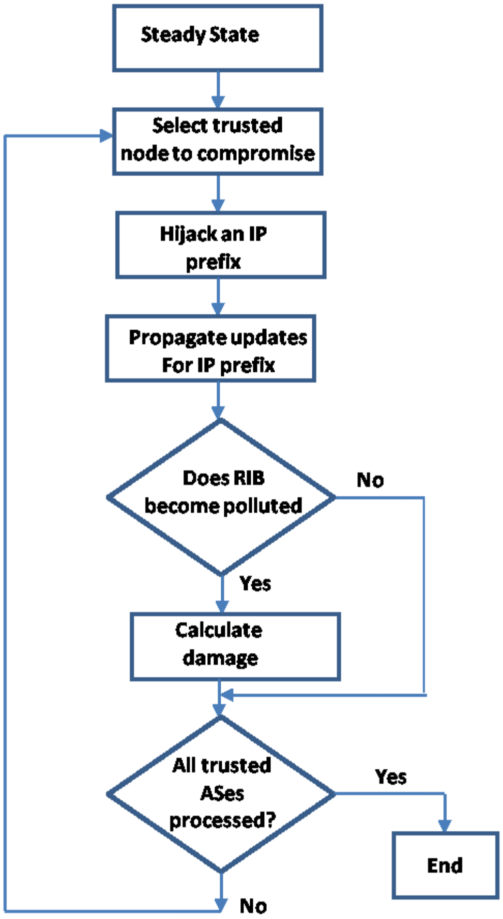
Steady State
Select trusted node to compromise
Hijack an IP prefix
Propagate updates For IP prefix
Does RIB become polluted
No
Yes
Calculate damage
All trusted ASes processed?
Yes
End
No

Figure 5-1: Simulation flowchart diagram for spoofed update messages.

Using the simulation model, we obtain results for both analytical model as well as the experimental models and we capture them in Figure 5-2. The analysis in Figure 5-2 assumes that there is only one malicious AS in the system. There are several observations which are discussed next.

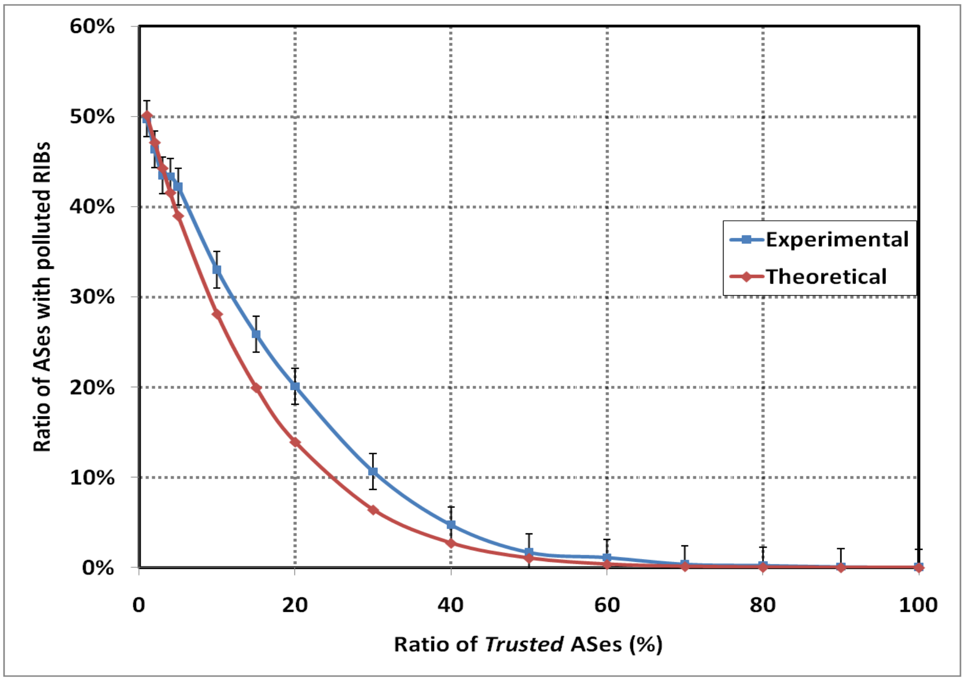

Figure 5-2: Ratio of ASes whose RIBs get polluted as a result of spoofed update messages.

As expected, the ratio of ASes with polluted RIBs declines as the number of *trusted* ASes is increased in the system. This is because the increased number of *trusted* ASes is able to detect the spoofed UPDATE messages and prevent them from traversing the network any further.

1. Note that with a very small number of *trusted* ASes, the ratio of ASes with polluted RIBs is ~50%. This is due to the fact (which was discussed earlier) that in a perfectly random topology and choice of ASes *m*, *v* and *a*, AS *a* has a 50% chance of being closer to AS *v* versus malicious AS *m*. Therefore, on average 50% of the ASes will end up with their RIBs polluted if there are no *trusted* ASes to verify signatures of spoofed BGP UPDATE messages.
2. As expected, when all the ASes in the network are *trusted* (100% ratio of *trusted* ASes), there are no ASes with polluted RIBs. However, it is very interesting to note that even if 60% of

the ASes in the network become *trusted*, the number of ASes with polluted RIBs is approaching 0. This is encouraging as it indicates that within the given topology and 60% of the ASes being *trusted*, the system is almost completely immune to IP prefix hijacking attacks from a malicious *trusted* AS in the network.

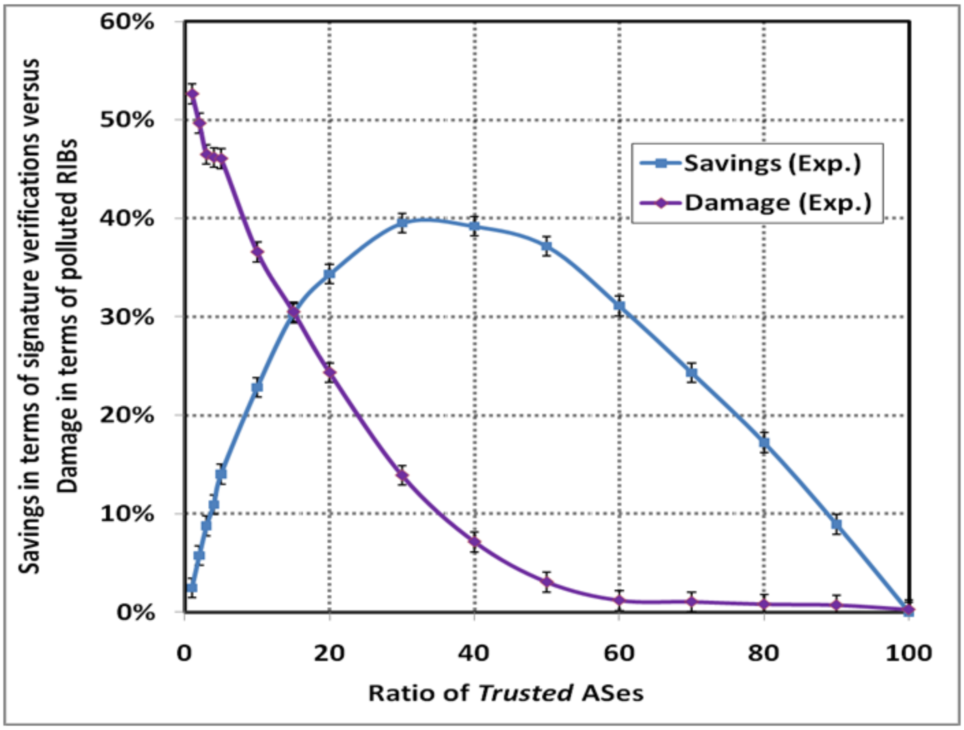

Figure 5-3: Ratio of damage as well as savings in signature verifications with *trusted* ASes.

3. We present both the savings and damage, in terms of number of signature verifications and the ratio of polluted ASes, respectively, in Figure 5-3. It is very interesting to note that the savings increase as the number of trusted ASes is increased in the system. However, the savings peak when the ratio of trusted ASes is approximately 35%. Beyond this ratio, the savings actually start to decline. This is an interesting phenomenon and is caused by the fact that as more ASes are made *trusted* in the network, more ASes end up performing signature verifications of the full AS-PATH, thereby more signature verifications. The benefit of *non-*

trusted ASes performing signature verification only up-to the last *trusted* AS in the AS-PATH start to decline once more than 35% of the ASes are made *trusted* in the network.

4. We observe that even though the savings peak when ratio of trusted ASes is 35%, there is still significant damage (approximately 10%) which can be performed by a malicious *trusted* AS. However, this damage declines to less than 1% when approximately 60% of the ASes are deployed as *trusted*. With 60% of the ASes as *trusted*, there is approximately 31% savings in terms of number of signature verifications. Also, the system is almost immune to IP prefix hijacking attacks (less than 1% damage) from a malicious *trusted* AS in the network
5. There is a close agreement between the results from the analytical model and the experimental model. We did investigate the small discrepancy between the two results and could attribute it to the fact that some ASes will always choose the malicious AS in some of their routes as the malicious AS happens to be in legitimate topological AS-PATH between the source AS and the destination AS.

5.3 Data-plane "Damage" Performed by a Malicious AS

In this section, we present an analytical formulation of the data plane "damage" performed by a malicious AS. As stated earlier, a malicious AS hijacks an IP prefix belonging to a legitimate AS *v* by announcing AS *v*'s IP prefix to the rest of the network via spoofed UPDATE messages. This causes RIB pollution in certain number of the ASes. We now study the impact of this malicious activity on the data traffic being sent in the network. If an AS *a* sends a data packet to AS *v* and this data packet ends up reaching malicious AS *m* instead, we consider this to be a successful attack.

Intuitively, one would think that the success of the data packet attack will be directly correlated to the ratio of ASes with corrupted RIBs. This is because the decision to forward traffic is based on the information stored in the RIB. We aim to study and prove this correlation via an original analytical model.

5.3.1 Analytical Model

We define damage *D* as the ratio of ASes whose traffic ends up in the malicious ASes instead of the true owner ASes of the IP prefixes. We model the path taken by a data packet that travels from a AS *s* to the destination (*v* or *m*) using the Markov chain depicted in Figure 5-4.

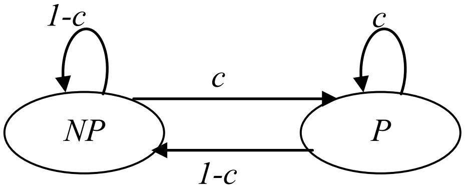

Figure 5-4: Path traversal of a data packet.

In Figure 5-4, state *NP* denotes a normal (or non-polluted) AS, whereas state *P* denotes a corrupted (polluted) AS. Once the data packet reaches a normal AS (state *NP*), there is a probability *c* that it will traverse a polluted AS (state *P*) next, and probability *1-c* that it will traverse another *NP* AS next. Similarly, once the packet reaches a polluted AS (state *P*), there is a probability *1-c* that it will traverse another polluted AS (state *P*) next, and probability *c* that it will traverse a *NP* AS next.

In order for the data packet to end up in the malicious AS *m*, instead of the true owner AS of the IP prefix, AS *v*, the data packet must be in state *P*, just before it reaches AS *m*. This is because if the data traffic is in state *NP*, then it will be directed towards AS *v* (righteous destination), instead and will not reach AS m. We can approximate damage, *D*, with the probability that the RIB of the last AS in the path (of length *d*) of the packet is polluted, and therefore will forward the packet to the malicious AS m. In reality, some data packets will traverse through AS m (and get counted towards damage), because AS m happens to be in legitimate topological AS-PATH between the source AS and the victim AS *v*.

$$D = Pr\ (X_{d-1} = P) \tag{5.3}$$

The marginal distribution $Pr\ (X_{d-1} = P)$ can be written using the initial distribution as follows:

$$\mathrm{P}\,r(X_{d-1} = P) = \sum\nolimits_{r \in S} p_{rP}^{(d-1)} * Pr\ (X_0 = r) \tag{5.4}$$

In Equation 5.4, *S* is the state space of the Markov chain, $p_{rP}^{(d-1)}$ is the probability of going from state *r* to state *P* in *d-1* transitions.

The *n*-step transition probabilities satisfy the *Chapman-Kolmogorov* equation (5.5), that for any *k* such that 0 < *k* < *n*,

$$p_{ij}^{(n)} = \ \sum\nolimits_{r \in S} p_{ir}^{(k)} * p_{rj}^{(n-k)} \tag{5.5}$$

Therefore, we can calculate the n-step transition probabilities as follows:

$$p^{(1)} = \begin{bmatrix} c & 1-c \\ c & 1-c \end{bmatrix} \tag{5.6}$$

By using the *Chapman–Kolmogorov* equation, we get:

$$p^{(2)} = \begin{bmatrix} c & 1-c \\ c & 1-c \end{bmatrix} \tag{5.7}$$

Therefore,

$$p^{(n)} = p^{(1)} = \begin{bmatrix} c & 1-c \\ c & 1-c \end{bmatrix} \tag{5.8}$$

Hence, we can calculate the $Pr\ (X_{d-1} = P)$ as:

$$P\,r(X_{d-1} = P) = \sum_{r \in \{P,N\}} p_{rP}^{(d-1)} * \Pr\,(X_0 = r) \tag{5.9}$$

$$P\,r(X_{d-1} = P) = p_{PP} * \Pr(X_0 = P) + p_{NP} * \Pr(X_0 = N) \tag{5.10}$$

$$P\,r(X_{d-1} = P) = c * c + c * (1 - c) \tag{5.11}$$

$$P\,r(X_{d-1} = P) = c \tag{5.12}$$

Hence,

$$D = c \tag{5.13}$$

As expected, the damage caused by a malicious AS is equal to the ratio of ASes with polluted RIBs due to IP prefix hijacking.

5.3.2 Simulation Model and Results

Figure 5-5 depicts the flowchart diagram for spoofed UPDATE messages and calculations pertaining to resulting damage in terms of attracting data traffic originally intended for rightful owner of the IP prefix. The sequence is as follows:

1. Once the entire routing table has reached steady state, a *trusted* AS is randomly selected as compromised and becomes malicious.
2. The AS hijacks a random IP prefix belonging to some other AS and starts advertising it in the network as if it owns it.

3. The IP prefix is propagated in the network and each AS uses the Path Selection Algorithm to accept or reject the spoofed UPDATE message.
4. From all other ASes in the network, we send data traffic to the hijacked IP prefix.
5. We inspect the compromised AS and determine the ratio of traffic it attracted. It is important to note that few ASes will forward the traffic to the compromised AS, not because of corrupt RIB, but because the compromised AS may already happen to be in the correct AS-Path.

Steps 1-5 above are repeated for all *trusted* ASes, and the ratio in step 6 is averaged over the total number of *trusted* ASes in the network.

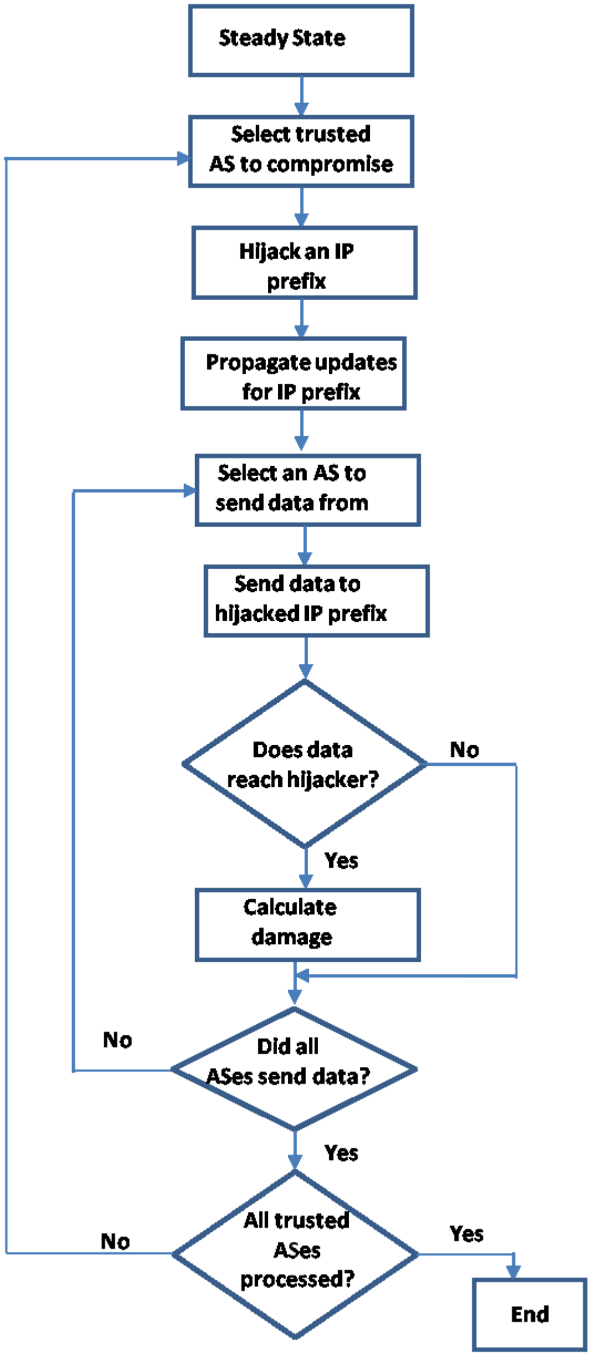
Steady State
Select trusted AS to compromise
Hijack an IP prefix
Propagate updates for IP prefix
Select an AS to send data from
Send data to hijacked IP prefix
Does data reach hijacker?
No
Yes
Calculate damage
No
Did all ASes send data?
Yes
No
All trusted ASes processed?
Yes
End

Figure 5-5: Flowchart diagram for spoofed UPDATE messages and damage calculation.

Using the simulation model, we obtain results for both analytical model as well as the experimental models and we capture them in Figure 5-6. Once again, the analysis in Figure 5-6 assumes that there is only one malicious AS in the system.

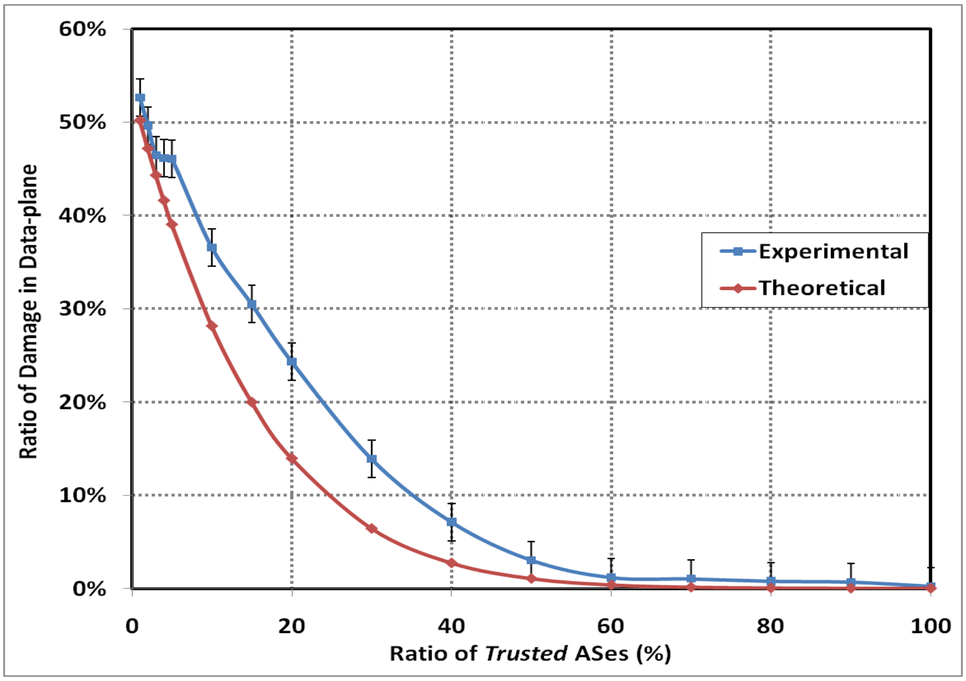

Figure 5-6: Ratio of damage in data-plane as a result of spoofed update messages.

As expected, the results for data plane damage ratio are very similar to the results for the ratio of ASes with polluted RIBs. This is because the decision to forward data traffic is based on the information stored in the RIB. There is reasonably close agreement between the results from the analytical model and the experimental model. In general, the damage is higher in simulation results as compared to results from analytical model. In the simulation model, all the traffic arriving at the malicious node is counted as damage. Some of this traffic may be legitimate traffic as the malicious AS may legitimately be present in the AS-PATH for many BGP UPDATE messages. This traffic, although legitimate, is counted as damage in the simulation model, since it is traversing or terminating on the malicious AS.

5.4 Damage Due to Multiple Malicious ASes

In our analysis so far, we have only studied damage caused by a single malicious AS in the system. In this section, we study damage impact due to simultaneous corruption of multiple *trusted* ASes in the network. For example, how does the security of the system degrade when more than 1 *trusted* AS in the network become malicious? Although highly unlikely due to the fact that *trusted* ASes would be carefully chosen and would have tight security procedures in place, it can happen in the case of a coordinated attack on multiple ASes.

For this analysis, we select a certain ratio of *trusted* ASes in the network, and study the impact on the ratio of RIB pollution if we gradually increase the number of *trusted* ASes participating in IP prefix hijacking. The assumptions made for the behavior of malicious ASes is as follows:

- Multiple ASes advertise the same IP prefix to make the network think that they individually own this prefix.
- When a malicious AS receives an UPDATE message containing an IP prefix that is being advertised by any of the other malicious ASes, the UPDATE message is accepted and forwarded to rest of the network.
- When a malicious AS receives an UPDATE message containing an IP prefix that is being advertised by the rightful owner of the IP prefix, the UPDATE message is rejected and dropped. The malicious ASes do not let the "correct" UPDATE message propagate beyond them.

We enhance the simulation model to incorporate the behavior above and document the results in Figure 5-7. We assume the number of colluding malicious *trusted* ASes varies from 1 to 5 and we repeated the experiment with ratio of *trusted* ASes set at 20%, 40% and 60%. From the results, it is obvious that as we increase the number of colluding *trusted* ASes, there is very significant damage happening in the network. It is interesting to note that with a higher overall ratio of *trusted* ASes in the network, the impact of colluding ASes is smaller. Considering the cases of 5 colluding ASes and comparing the damage with 20% overall *trusted* ASes and 60% overall *trusted* ASes, the damage drops from approximately 68% to 19%. This is not unexpected as with a higher ratio of *trusted* ASes, the impact of colluding ASes is more localized and contained. Even though a damage of 19% at 60% trust ratio is very high and potentially unacceptable, we believe that the case of 5 *trusted* ASes becoming malicious and colluding to attack the network is an extreme and highly unlikely case.

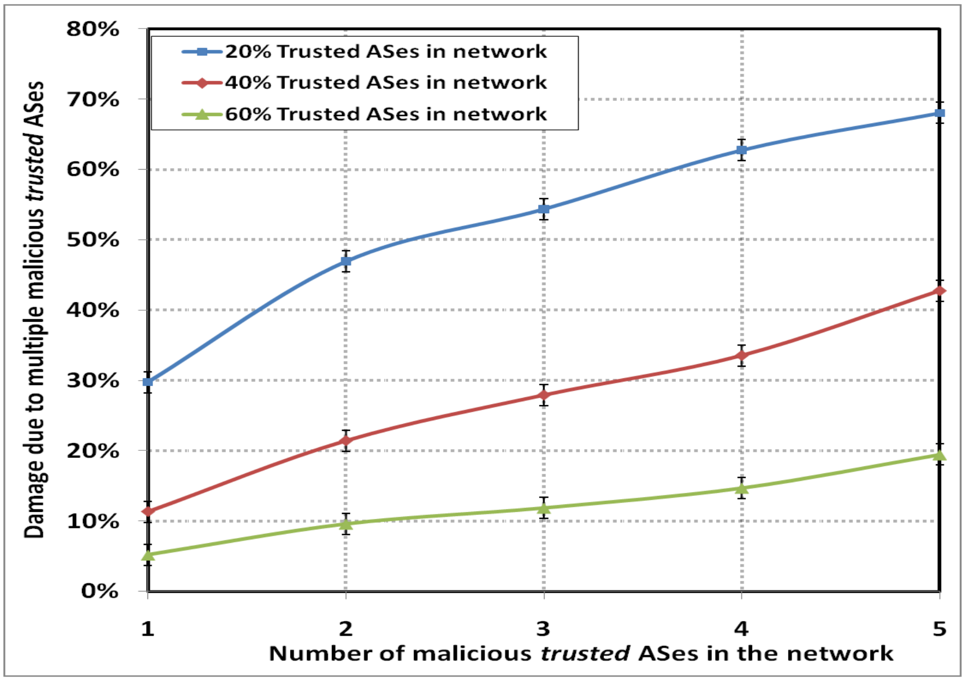

Figure 5-7: Damage due to colluding malicious *trusted* ASes.

5.5 Conclusion

In this chapter, we discussed control plane and data plane security aspects of proposed C-BGP. We developed analytical models to seek the extent of control plane and data plane damages when one of the *trusted* ASes becomes malicious. We also enhanced our simulation model to incorporate control plane and data plane damage analysis and compared results. We determined that there is a significant incentive to deploy 60% of the ASes as *trusted* considering the results indicating that there is 31% savings in terms of number of signature verifications and the system is almost immune to IP prefix hijacking attacks (less than 1% damage) from a malicious *trusted* AS in the network. We also studied damage in data plane and confirmed that it is very similar to the damage caused in control plane. We also considered the impact of multiple *trusted* ASes colluding to launch an IP prefix hijacking attack and saw the network performance (in terms of ability to limit damage) degrade significantly. This is not unexpected, as C-BGP builds on top of concept of trust.

Chapter 6

A Hybrid Cryptosystem to Secure BGP

6.1 Introduction

In this chapter, we present a hybrid cryptosystem to expedite repeated and frequently occurring signature verifications happening in the proposed C-BGP architecture. The following factors motivate the need for such hybrid cryptosystem.

- We recognize that it may not be possible for all the ASes in the Internet to migrate to a system where all ASes are performing signature verifications, as this would most likely require a hardware upgrade in almost all location. It would be more practical to minimize the impact on smaller ASes by deferring signature verification operations to major ASes only in the network.
- For smaller ASes, it may be low enough risk to accept BGP UPDATE messages without real-time verifications of the signatures. As long as the signatures are verified in due time, any impact to network performance during this time may be considered acceptable. It is not unreasonable to think that initial deployments may rely on signature verifications between major ASes only.

- Recall that signature verifications on an asymmetric-key based system is a slow process (see Section 2.4.1). Hence, the time taken for overall network convergence may not be acceptable. To bring the overall network convergence time down, a system consisting of both asymmetric and symmetric verifications can be considered.

6.2 Hybrid Approach

In this section, we briefly present the architecture of a new extension to C-BGP that is distinguished by three aspects, which we believe, are unique and offer a pragmatic compromise in terms of low overall computational overhead and an acceptable level of security as compared to existing BGP. These include:

- Concept of *trusted* AS in the network
- Use of hybrid cryptosystem in the network to expedite repeated and frequently occurring signature verifications.
- Concept of delayed or deferred verification of the UPDATE messages.

We note that a symmetric approach to secure BGP has been proposed in [BEZ11]. The authors used the Square Grid scheme to reduce the number of symmetric keys required to exchange the route updates. However, the solution proposed requires each intermediate AS to insert a digital signature of the update using the shared key with every AS that might receive the update. We believe that this model has limitations in that the signing AS cannot predict the AS-PATH.

6.2.1 Hybrid Cryptosystem Approach

The new extension of C-BGP uses a hybrid cryptosystem and the concept of *trusted* ASes in the network to achieve an acceptable level of security and improved computational performance as compared to S-BGP. In this hybrid cryptosystem model, both asymmetric and symmetric cryptography are employed to accomplish different tasks in the network. We call this model Hybrid Credible BGP (HC-BGP). Recall that asymmetric key algorithms are typically hundreds to thousands times slower than symmetric key algorithms [NAR08]. Therefore, we propose that the asymmetric key cryptosystem be deployed to communicate with *trusted* ASes; and the symmetric key cryptosystem approach be used to communicate between *non-trusted* ASes and their *trusted* AS neighbors. We assume that an AS, t, is a *trusted* neighbor of an AS, n, if and only if there exists a path, p, between the AS, n, and the *trusted* AS, t, where there is no other *trusted* AS in path p. Each *non-trusted* AS shares its symmetric private key with all of its neighboring *trusted* ASes. This key is not propagated beyond the first set of neighboring *trusted* ASes. The shared symmetric key can be exchanged offline, or using a secure communication protocol that uses the public keys of the *trusted* ASes. It is assumed that there exists a mechanism for each *non-trusted* AS to securely reach neighboring *trusted* ASes in the network.

6.2.2 Real Time and Delayed Verifications

Each *non-trusted* AS can operate in one of two modes: "Real-time Validation (RV)" mode or "Deferred Validation (DV)" mode. It is up to the network operator to configure the mode of operation. In RV mode, upon reception of an update, the *non-trusted* AS immediately requests a validation of the *non-trusted* portion of the AS-PATH from one of its neighboring *trusted* ASes. The *trusted* AS processes the UPDATE message and sends VALID/INVALID message to the requesting AS. In DV mode, the AS waits until the update is received by a *trusted* AS in the downstream path. The *trusted* AS confirms/denies the validity of the update to the *non-trusted* AS. If the update reaches the last AS in the path without reaching a *trusted* AS in the path, the last AS sends a request to neighboring *trusted* ASes to validate the *non-trusted* portions of the AS-PATH.

For this purpose two messages are introduced by BGP. The first message type, called “Validation Query (VQ) Message”, is used by *non-trusted* ASes to query *trusted* ASes for validity of the message. Another message type, “Validation Response (VR) Message” is introduced to communicate the validity of the UPDATE message from the *trusted* ASes to the *non-trusted* ASes.

6.2.3 Detailed Description

The mode of operation of the protocol depends on the type of the AS (*trusted*/*non-trusted*) and the verification mode of the *non-trusted* ASes. Figure 6-1 depicts the messages exchanged to perform the protocol operations.

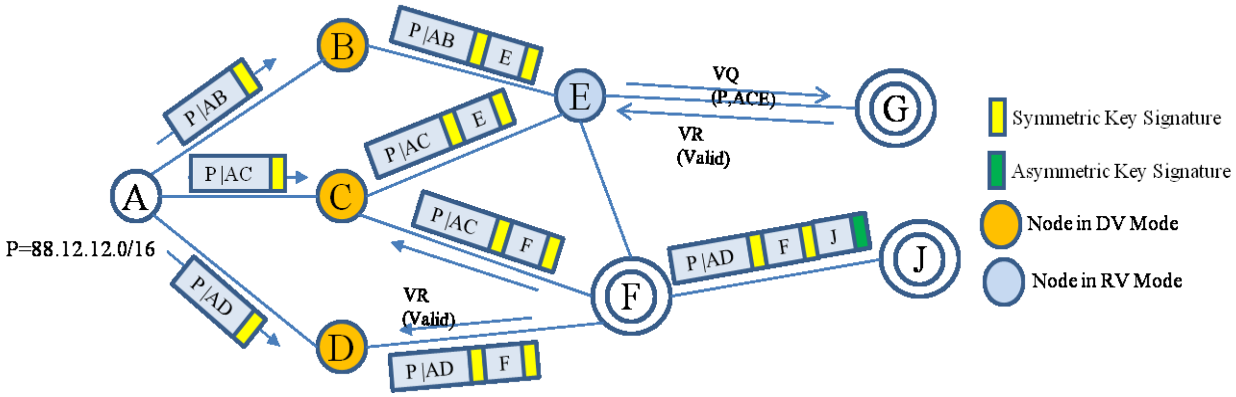

Figure 6-1: Traversal of HC-BGP UPDATE messages.

1. As illustrated in Figure 6-1, *non-trusted* ASes B, C and D act in deferred mode. Upon reception of route update from originating AS A, they perform the following actions:
2. Skip signature verifications on the incoming message,

3. Install the prefix in the forwarding table,
4. Add the next AS information to the AS-PATH,
5. Sign the resultant UPDATE message,
6. Forward the UPDATE message to the next AS,
7. Start a RESPONSE WAIT timer, with configurable timer value, to wait for a "Validation Response Message" from the first *trusted* AS in the downstream path.

Upon reception of the update, *trusted* AS F verifies the embedded signature of all ASes (*trusted* or *non-trusted*) in the AS-PATH. Recall that *non-trusted* ASes share symmetric keys with their immediate *trusted* ASes. Depending on whether the verification is successful or not, the *trusted* AS sends a valid or invalid "Validation Response Message" to each of the *non-trusted* ASes between the last *trusted* AS and itself in the AS-PATH (in this case C and D). Afterwards, the AS appends the next AS, J in this example, and signs the full path with its private key and forwards it to the next AS.

On the *non-trusted* ASes C and D, if the RESPONSE WAIT timer expires and the "Validation Response Message" is not received from the neighboring *trusted* AS, the prefix is withdrawn from the forwarding table and a WITHDRAW message is sent out to the network. If the "Validation Response Message" is received and it indicates that the UPDATE was "INVALID", the prefix is withdrawn from the forwarding table and a WITHDRAW message is sent out to the network. At last, if the "Validation Response Message" is received and it indicates that the UPDATE was "valid", the entry that resulted from the update remains valid.

In RV mode, when *non-trusted* AS E receives a BGP UPDATE message, it will perform following actions:

1. Send a VQ message with the contents of the UPDATE message to the neighboring *trusted* AS in order for it to confirm the validity of the UPDATE message. The symmetric keys between the *trusted* and *non-trusted* ASes are used to authenticate the sender and secure the content.
2. Start a RESPONSE WAIT timer (with configurable timer value) to wait for a "Validation Response Message" from the neighboring *trusted* AS.
3. When a *trusted* AS receives the VQ message, it performs verification of the AS portion of *non-trusted* ASes that is not verified. Depending on whether the verification is successful or not, it sends a valid or invalid "Validation Response Message" to the querying *non-trusted* AS.
4. If the "Validation Response Message" is received and it indicates that the UPDATE was "INVALID", the UPDATE message is rejected and dropped.
5. If the "Validation Response Message" is received and it indicates that the UPDATE was "VALID", the UPDATE message is used to install the prefix in the forwarding table of the AS.
6. Last, *non-trusted* AS E adds the next AS information to the AS-PATH, signs the resultant UPDATE message, and forwards the UPDATE message to the next AS (AS G in this example).

The Deferred Validation (DV) mode introduces flexibility in the protocol and allows high protocol convergence speed by deferring the verification overhead to post-convergence timeframe. It is conceivable that this mode offers a "window of opportunity" for attackers to inject a false UPDATE message; this window can be reduced by configuring a very small RESPONSE WAIT timer value. This mode is particularly attractive in deployment scenarios where operators demand high performance at the expense of small timing windows of exposure. Considering that majority of incorrect BGP UPDATE messages in the Internet today are not caused by attackers, instead, they are caused by the mis-configuration of BGP operators; this mode would be very attractive compromise.

The Real Time Validation (RV) mode offers strict security whereby no UPDATE message is installed in RIB until it has been verified by the *trusted* AS. The verification operation is expected to be quick since the *trusted* AS will only verify the AS-PATH between itself and the last *trusted* AS in the AS-PATH. Also, since symmetric key operation is deployed, the verification operation is expected to be fast. It is conceivable that a *non-trusted* AS may bulk few UPDATE messages in the VQ message to reduce the number of VQ messages being exchanged in the network. Similarly, the *trusted* AS may decide to bulk several responses in a single VR message to cut down on the number of VR messages being exchanged.

6.3 Analytical Model

In this section, we develop theoretical formulas to calculate the average number of asymmetric and symmetric verifications performed.

6.3.1 Average Number of Asymmetric Verifications

We denote by A_SigV_n the average number of asymmetric signature verifications performed by the AS at hop n.

The AS will perform one asymmetric verification per *trusted* AS in the traversed path. Since there is a ratio *x* of *trusted* ASes in the network, the number of *trusted* ASes in the AS-PATH is equal to $x * n$. Therefore the number of signature performed by AS at hop *n* is expressed as:

$$A\_SigV_n = x * n \tag{6.1}$$

Therefore, the average number of asymmetric signature verifications performed by each *trusted* AS per IP prefix is:

$$avgAsymSigVerificationsPerPrefix = \sum_{n=1} \frac{|\alpha_n| * A\_SigV_n}{x * N} \tag{6.2}$$

With α_n is the set of ASes that are *n* hops away from the origin of the update.

Assuming that each AS advertises one IP prefix, we can write the general formula for the average of asymmetric signature verifications performed by each *trusted* AS in the presence of a percentage *x* of *trusted* ASes:

$$avgAsymSigVerifications = \sum_{n=1} n * |\alpha_n| \tag{6.3}$$

6.3.2 Average Number of Symmetric Verifications

We denote by S_SigV_n the average number of signature verifications performed by the AS at hop *n*.

The number of **symmetric** verifications required to process an update is equal to the number of *non-trusted* ASes in the AS-PATH. This quantity can be expressed as:

$$S\_SigV_n = (1 - x) * n \tag{6.4}$$

Therefore, the average number of **asymmetric** signature verifications performed by *trusted* each AS per IP prefix is:

$$avgSigSymVerificationsPerPrefix = \sum_{n=1} \frac{|\alpha_n| * S\_SigV_n}{x * N} \tag{6.5}$$

With α_n is the number of ASes that receives the update with an AS-PATH length equal to *n*. Assuming that each AS advertises one IP prefix we can write the general formula for the average of asymmetric signature verifications performed by each AS in the presence of a percentage *x* of *trusted* ASes:

$$avgSymSigVerifications = \sum_{n=1} |\alpha_n| * \frac{n*(1-x)}{x} \quad (6.6)$$

6.3.3 Total Cost of Verifications

Equations (6.3) and (6.6) represent the average cost of asymmetric and symmetric signature verifications, respectively. The total cost of verifications expresses the average amount of processing time required to verify both types of signatures at the level of a *trusted* AS. If we assume ρ is the ratio of asymmetric signature verifications time units to symmetric verifications time units, we can represent the total cost of verifications as:

$$Verification\_Cost = avgSymSigVerifications + \rho * avgAsymSigVerifications \quad (6.7)$$

By substituting equations (6.3) and (6.6) for $avgAsymSigVerifications$ and $avgAsymSigVerifications$ respectively, we get:

$$Verification\_Cost = \sum_{n=1} n * |\alpha_n| * \left[1 + \rho * \frac{(1-\mathrm{x})}{\mathrm{x}}\right] \quad (6.8)$$

6.4 Messaging Overhead of Real-time Validation Mode

The real-time validation mode requires that each *non-trusted* AS to request the verification of the update from a neighboring *trusted* AS. Therefore, the number of requests sent is equal to the number of updates processed (received) by *non-trusted* ASes. This quantity can be expressed as:

$$message\_overhead\_perPrefix = \sum_{n=1} |\alpha_n| * (1 - x) \quad (6.9)$$

Therefore, assuming that each AS advertises one IP prefix, the average number of messages sent by *non-trusted* ASes to validate BGP UPDATE messages is expressed as:

$$message\_overhead = N * \frac{message\_overhead\_perPrefix}{N*(1-x)} \tag{6.10}$$

This can be simplified as:

$$message\_overhead = \sum_{n=1} |\alpha_n| \tag{6.11}$$

6.5 Performance Results

In this section, we present data collected with the experimental measurement infrastructure described in the previous section. As noted in Section V, AS-Path length serves as the only metric for route preference. We simulated a BGP network with more than 2,000 ASes and BGP selection algorithm is based solely on the shortest path. We introduced gradually 1% of *trusted* ASes in the network and measured following metrics to evaluate the performance of the new scheme.

- Number of symmetric verifications performed by *trusted* ASes.
- Number of asymmetric verifications performed by *trusted* ASes.
- Average number of validation queries performed by *non-trusted* ASes in real-time mode.

Using metrics above, we also derived following parameters that are of interest to this study.

- Average unit of time spent to perform total number of signature verifications in C-BGP and HC-BGP (this was derived from the metrics above).
- Ratios of time spent performing signature verifications assuming that HC-BGP relied entirely on asymmetric verifications (even between *trusted* and *non-trusted* ASes).

All the measurements were performed while the network exchanged announcements to reach the steady state. *Trusted* ASes are selected based on the highest number of peers. Figure 6-2 shows the average number of symmetric signature verifications performed by each *trusted* AS. The cost of verifications is decreasing significantly (almost exponentially) as the number of *trusted* ASes increases. This can be explained by the fact that the burden of verifications is shared by more ASes as an increasing number of them become *trusted*. On the other hand, as shown in Figure 6-3, the average number of asymmetric signature verifications performed by *trusted* ASes is almost constant as the ratio of *trusted* ASes is increased. This is because as we increase the number of *trusted* ASes in the system, there will be more asymmetric signature verifications that will be needed to be performed. In the simulation model, we assume that each *trusted* AS has to verify the embedded signature of the entire AS-PATH. It is important to note that *non-trusted* ASes are offloaded from these expensive cryptographic verifications.

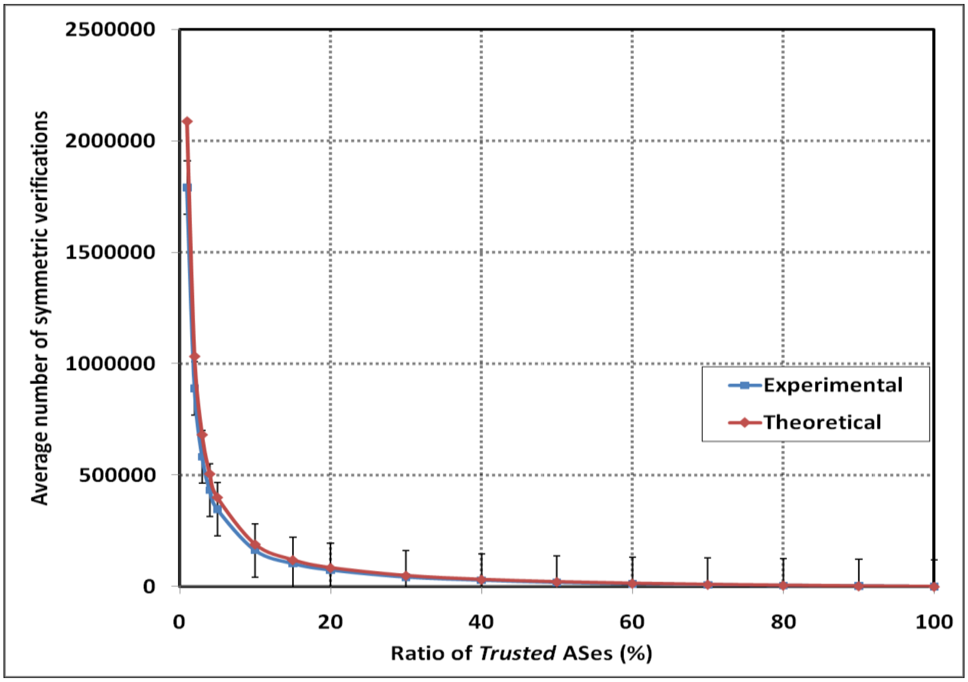

Figure 6-2: Average number of symmetric verifications performed by *trusted* ASes.

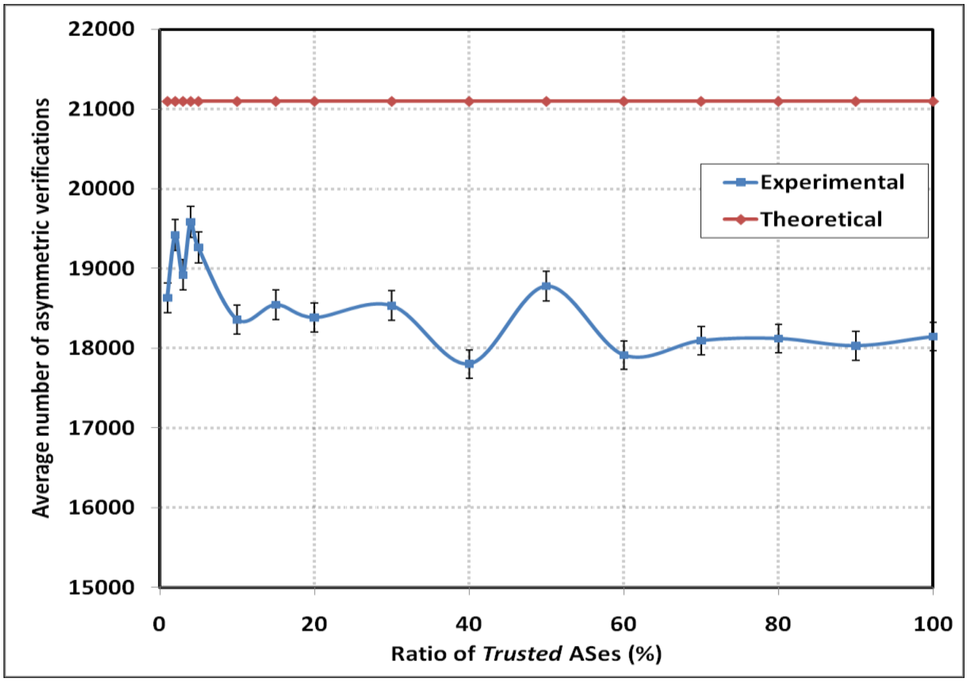

Figure 6-3: Average number of asymmetric verifications performed by *trusted* ASes.

HC-BGP requires each *non-trusted* AS to delegate verification of UPDATE messages to *trusted* ASes. As discussed previously, this delegation can be done in two modes: real-time verification or deferred Verification. We simulated real-time mode of verification and plotted per *non-trusted* AS messaging overhead in Figure 6-4. As expected, the overhead, on per *non-trusted* AS basis remains approximately constant. This is because regardless of the number of *trusted* ASes in the system, each *non-trusted* AS will need to process same number of UPDATE messages and send them to the closest *trusted* AS to verify.

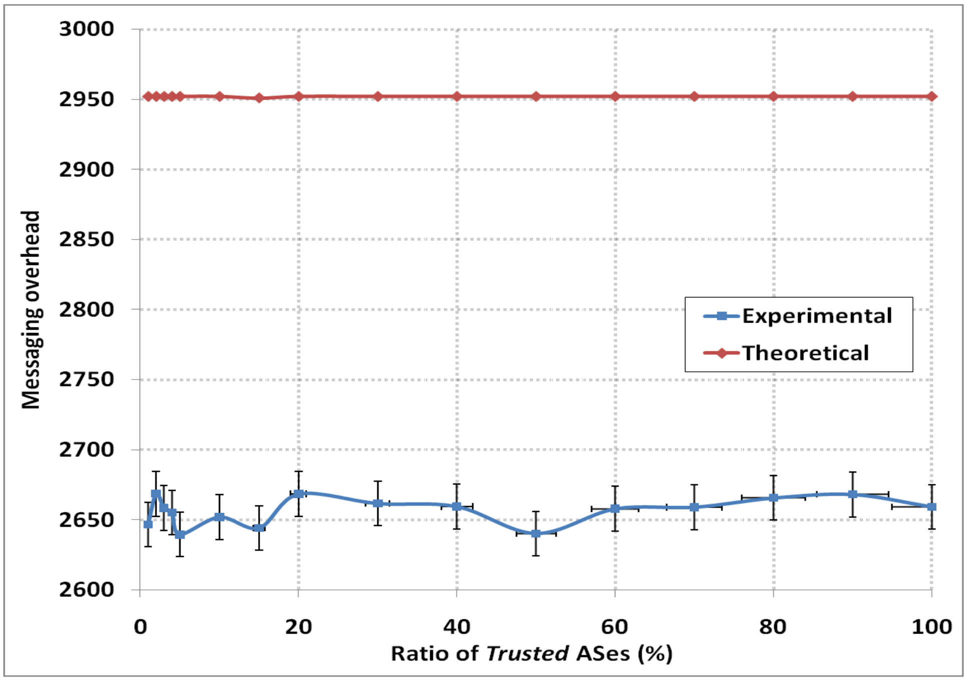

Figure 6-4: Real-time mode messaging overhead per *non-trusted* AS.

As discussed earlier, signature verifications on an asymmetric-key based system is much slower than the same on symmetric-key based system. Since C-BGP is an asymmetric-key based system and HC-BGP is a hybrid system, we were interested in determining how the two systems will compare in terms of linear amount of time taken to complete the above discussed average number of signature verifications. While it would not truly give us network convergence times as these operations happen in parallel, it would be an interesting comparison nonetheless. We wanted to determine average units of time spent to perform total number of signature verifications in C-BGP and HC-BGP. For this analysis, we assumed that asymmetric key operations are a factor of r slower than symmetric key verification operations. The values of ρ have ranged from 100 in software based systems to 1000 or more in hardware based systems [RSA91]. To get fair comparison between C-BGP and HC-BGP, we assumed that each symmetric key operation in HC-BGP would take 1 unit of time. To convert asymmetric key operations into same units of time, we chose to convert average number of asymmetric verification operations performed in C-BGP and HC-BGP by multiplying them by ρ factor. For HC-BGP, we summed the time unit values obtained from symmetric and asymmetric key operations. The resulting values are plotted in Figure 6-5 for ρ = 1000 and in Figure 6-6 for ρ = 100. Both the graphs reveal that HC-BGP will take many more units of time to complete signature verification processing. To be precise, with ρ = 1000 and ratio of trusted ASes at 20%, the ratio between the time unit numbers for C-BGP and HC-BGP is approximately 65% (see Figure 6-5). Similarly, with ρ = 100, same ratio is 62%. This clearly indicates that, on average, there are more asymmetric signature verifications required in HC-BGP as compared to C-BGP. This is because in C-BGP, both *trusted* and *non-trusted* ASes are performing signature verifications and *non-trusted* ASes have far fewer signature verifications to perform as they only verify the signatures up-to the last trusted AS in the AS-PATH. However, we must recall that in HC-BGP, there is no verification done whatsoever by *non-trusted* ASes, which is a significant incentive for ease of deployment.

With the results indicating that the total units of time spent verifying signatures is significantly higher in HC-BGP than C-BGP, an interesting question which can be posed is whether it is worth the extra complexity to deploy a hybrid crypto system? Why not discard symmetric cryptography and borrow the concept of offloading *non-trusted* ASes and apply it to C-BGP? Essentially, have a C-BGP system whereby only trusted ASes perform signature verifications using asymmetric cryptography. We performed signature verification time analysis for such a system and compared it with HC-BGP. Not unsurprisingly, the time units needed to verify all the signatures in this *modified* C-BGP system were significantly higher than in HC-BGP.

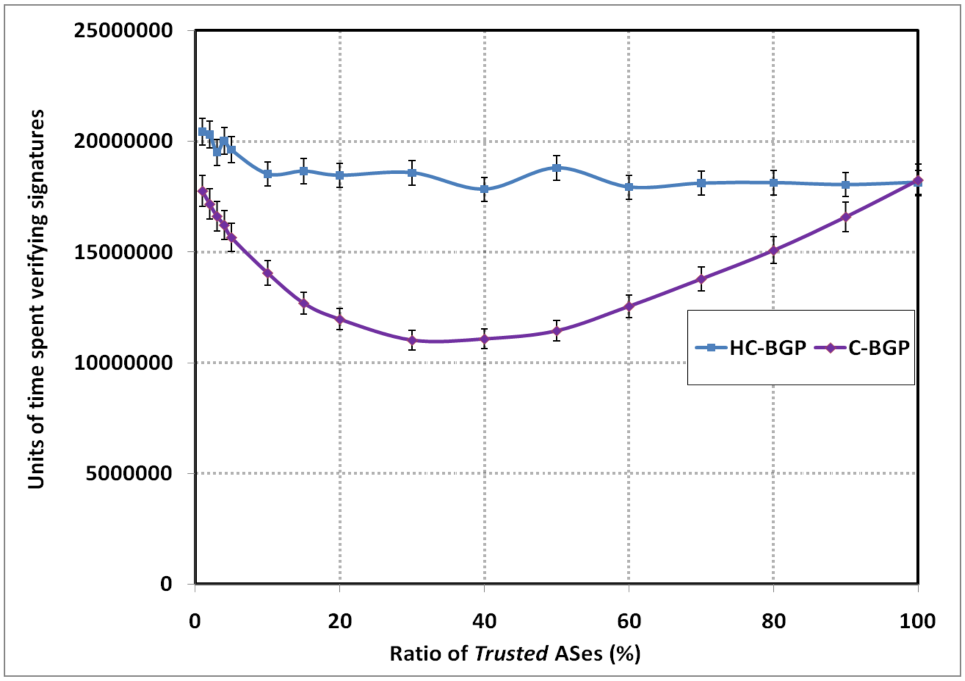

Figure 6-5: Average units of time spent verifying signatures with C-BGP and HC-BGP and ρ = 1000.

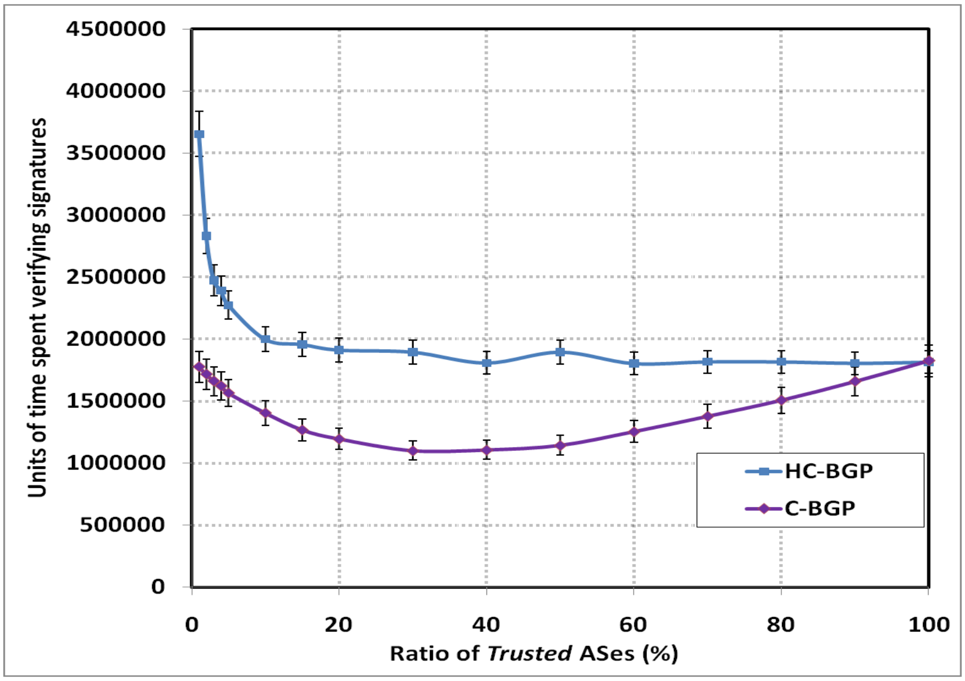

Figure 6-6: Average units of time spent verifying signatures with C-BGP and HC-BGP and ρ = 100.

In Figure 6-7, we plot the ratio of the time units taken by the modified C-BGP and HC-BGP. It is interesting to note that at low ratio of *trusted* ASes, HC-BGP performs much better. The impact due to symmetric verifications is much evident when there is a low ratio of trusted ASes. As the ratio of trusted ASes increases, both systems perform similarly. This is because there are far less symmetric verifications happening with high ratio of *trusted* ASes. We can conclude that with low ratio of trusted ASes, there is significant value in deploying hybrid cryptosystem.

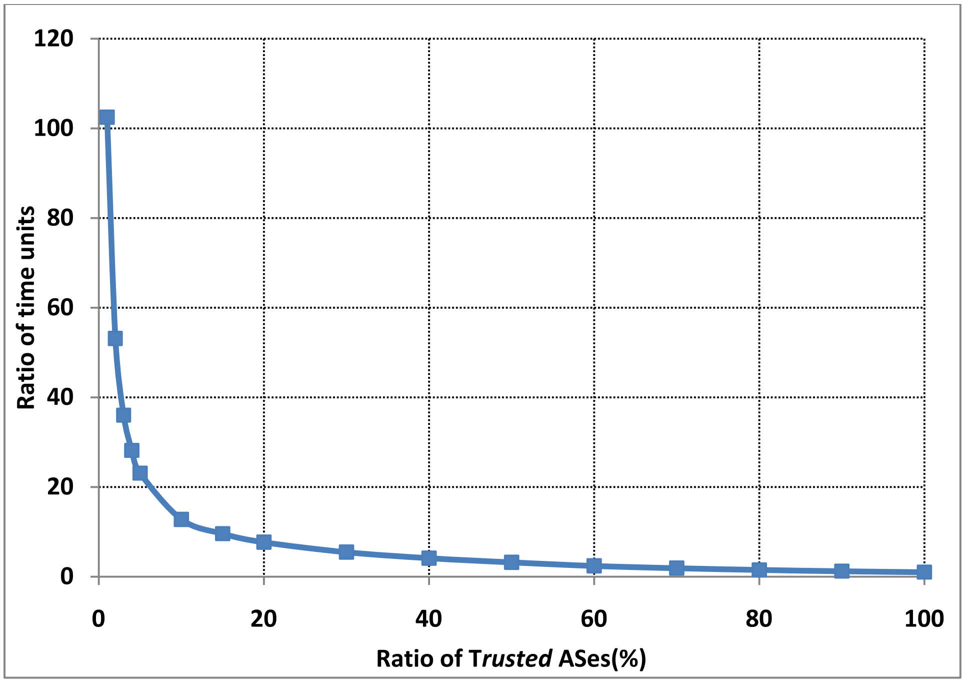

Figure 6-7: Ratio between modified C-BGP and HC-BGP of time units to perform verifications.

6.6 Conclusion

Recognizing the importance of lessening the burden on smaller ASes to ease adoption, we proposed HC-BGP which is based on a hybrid cryptosystem benefiting from flexible asymmetric cryptography and efficient symmetric cryptography. We developed an analytical model and a simulation model to study the number of symmetric and asymmetric verifications needed in HC-BGP. Indeed, by combining the use of a hybrid cryptosystem and the concept of *trusted* ASes in the network, we were able to help the smaller, *non-trusted,* ASes by offloading signature verifications to *trusted* AS only. This is a significant benefit as it helps in transitioning the network to a more secure network by upgrading few ASes to a *trusted* status and allowing them to perform signature verifications. Also, the use of symmetric cryptography between *trusted* and *non-trusted* ASes means that overall system performance will improve, depending on the ratio of *trusted* ASes in the system.

Chapter 7

Conclusions and Future Research

7.1 Concluding Remarks

In this chapter, we present a review of our research contributions and concluding remarks, and propose some directions for future work.

Despite several proposals already put forward to secure BGP, securing BGP still remains a very active research topic in the research community. Any proposal for securing BGP must be incrementally deployable and must only add incremental changes to BGP. Since the Internet's operations remain highly dependent on seamless and uninterrupted operation of BGP, any system that calls for wholesale changes to BGP will be very difficult to deploy. The sensitivity of Internet's operations to BGP was exposed from the incident which happened on August 27, 2010 [NEU11] when RIPE NCC's Routing Information Service (RIS) was involved in an experiment using standards compliant optional attributes in BGP. The experiment caused a massive increase in routing instability. Up to 1.4% of the Internet was affected by instability around the time of the experiment.

In this thesis, we initially studied the performance of IDPF filters in the presence of malicious BGP UPDATE messages. IDPF filters are constructed based on routes created by BGP UPDATE messages and these routes are used to determine whether data traffic should be accepted or not. The basic premise behind IDPF is that if a single-path routing scheme is assumed and commercial relationships among ASes are considered, there is a small set of the number of *feasible* paths that can be used is exactly one single path *p(s, d)* between source AS *s* and destination AS *d*. Therefore, it is acceptable to discard any incoming packet with source address *s* and destination address *d* not in *p(s, d)*. However, if malicious ASes hijack IP prefixes and inject invalid UPDATE messages in the network, the performance of IDPF filters suffers heavily. We propose simple extensions to existing BGP which can be used to validate BGP UPDATE messages and discard any invalid messages. Using these extensions, we demonstrated that performance of IDPF filters improves significantly when under attack from malicious ASes. We concluded that these extensions were necessary to protect IDPF from setting up bogus filters.

We have proposed and studied C-BGP in detail, which introduced the idea of *trusted* ASes in the network. We have developed original and detailed analytical and simulation models to analyze the impact of trust on the number of signature verification operations in the presence of a (random) percentage of *trusted* ASes in the network. The study has shown that 20% of *trusted* ASes in the network can reduce the number of AS-PATH verifications by almost 33%. Similarly, the average number of IP prefix validations is reduced by 60% when 20% of the ASes are *trusted*. Also, the average number of public keys is reduced by 60% when 20% of the ASes are *trusted*. We have determined that there is a very significant incentive to deploy 60% of the ASes as *trusted* considering the results indicating that there is 31% savings in terms of number of signature verifications and the system is almost immune to IP prefix hijacking attacks (less than 1% damage) from a malicious *trusted* AS in the network. Also, we have considered the impact of multiple *trusted* ASes colluding to launch an IP prefix hijacking attack and observed significant degradation in network performance (in terms of ability to limit damage). This was largely not unexpected.

Considering Internet topology and its sensitivity to wholesale changes in BGP, we conclude that C-BGP is a pragmatic and incrementally deployable scheme which will provide acceptable level of network security to the network at an acceptable cost (in terms of number of signature verifications).

We have also proposed and discussed a hybrid cryptosystem to secure BGP. HC-BGP benefits from flexible asymmetric cryptography and efficient symmetric cryptography. By combining the use of a hybrid cryptosystem and the concept of *trusted* ASes in the network, we were able to help the smaller, *non-trusted,* ASes by offloading signature verifications to *trusted* AS only. This is a significant benefit as it helps in transitioning the network to a more secure network by upgrading few ASes to a *trusted* status and allowing them to perform signature verifications. We developed analytical as well as simulation models to study the number of symmetric and asymmetric verifications needed in HC-BGP. By combining the use of a hybrid cryptosystem and the concept of *trusted* ASes in the network, we were able to help the smaller, *non-trusted,* ASes by offloading signature verifications to *trusted* AS only. It can be concluded that the use of symmetric cryptography between *trusted* and *non-trusted* ASes results in improvement of overall system performance, depending on the ratio of *trusted* ASes in the system.

7.2 Future Research

The schemes proposed in this thesis can be further studied and extended in multiple dimensions as follows:

1. In the context of C-BGP and HC-BGP, it would be important to study their performance with new sets of constraints factored into the analytical and simulation models. We could incorporate commercial relationships between ASes into the topology to influence propagation of BGP UPDATE messages. In this thesis, we assumed a flat topology and allowed BGP UPDATE messages to propagate through all egress interfaces. By accounting for commercial relationships between ASes, BGP UPDATE messages will be forced to follows AS-PATHs that are different from our analysis and may produce different sets of results.
2. In our security analysis for C-BGP, we had chosen different random locations for *trusted* ASes. Given a flat topology or otherwise, we could study different particular locations of *trusted* ASes in the network, which if becomes malicious, will cause more damage as compared to other locations. This is an interesting question as it may yield some non-obvious locations for *trusted* ASes, especially if commercial relationships between ASes are taken into account.
3. In our analysis, we considered impacts of prefix hijacking on control and data planes in C-BGP and HC-BGP. There are other types of attacks which can also be considered for both analytical and simulation models. Some of these are discussed in [GOL10].
4. Currently, in C-BGP, both *trusted* and *non-trusted* ASes perform signature verifications. In HC-BGP, only *trusted* ASes perform verifications and notify the *non-trusted* ASes of the result. Another approach could be to consider a different system using partially signed BGP UPDATE messages only. The approach would be to only have few ASes (*trusted* or otherwise) sign and verify BGP UPDATE messages. Depending on the location of these ASes and success against common types of attacks, such a system would be very attractive as

it would conceivably be easier to deploy than a system requiring all ASes to participate in some way or another.

5. The research can also be extended to study initial and incremental deployment models of the proposed schemes in current Internet topology. The parameters of interest would be the performance impact on convergence due to extra signatures needed as well as the security gains made by the initial and incremental deployment.
6. Another important angle to study is the performance of the proposed schemes in terms of convergence times in the event of link failures or node reloads. It would be very interesting to understand that latency added to overall node and network convergence due to the overhead of signature validations and extra signaling, when network experiences failures.
7. This research can also be extended to evaluate the use and impact of secure multiparty digital signature schemes. This scheme enables a receiving AS to quickly decode an update message signed sequentially by multiple ASes and verify the authenticity as well as the order of signatures. The study can evaluate performance, in terms of processing speed and memory requirements, of multiparty digital signature schemes in the context of C-BGP and HC-BGP. The analysis could measure the computational overhead on the routers' CPU, the number of cryptographic operations required to sign and verify an update, and the total time required to reach the steady state.

Bibliography

[AIE03] W. Aiello, J. Ioannidis, and P. McDaniel. “Origin authentication in interdomain routing”. *In Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS 2003)*, pp. 165–178, Oct. 2003.

[AIE06] W. Aiello, K. Butler, and P. McDaniel. “Optimizing BGP Security by Exploiting Path Stability”. *In proceedings of the 13th ACM Conference on Computer and Communications Security (CCS 2006)*, pp. 298-310, Oct. 2006.

[AYR06] I. C. Avramopoulos and J. Rexford. “Stealth Probing: Efficient Data-Plane Security for IP Routing”. *In Proceedings of USENIX Annual Technical Conference, General Track'2006*. pp. 267-272, May 2006.

[BAK72] B. Baker and R. Shostak, “Gossips and Telephones”. Discrete Mathematics, No. 2, pp. 191–193, 1972.

[BAL07] H. Ballani, P. Francis, and X. Zhang. “A study of prefix hijacking and interception in the Internet”. *In Proceedings of the ACM SIGCOMM 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM 2007)*, pp 265-276, Aug. 2007.

[BAR06] E. Barr, D. DeFigueiredo, M. Nicholes, S. M. Shih Ming Tseng, C. Chuah, B. Mukherjee, and S. F. Wu. “Descartes BGP: A conflict detection and response framework for inter-domain routing”. Department of Computer Science, University of California, Davis, Tech. Rep., 2006.

[BAT08] T. Bates, P. Smith, and G. Huston, BCIDR Report for 30 January 08. Online, Mar. 2008. Available from http://www.cidr-report.org/

[BEL03a] S. Bellovin. “SBGP-Secure BGP”. *In Presentation Archives of the North Americans Network Operators' Group Meeting (NANOG28)*, June 2003.

[BEL03b] S. Bellovin, M. Leech, and T. Taylor. “ICMP Traceback Messages”. IETF Draft, Feb. 2003, http://tools.ietf.org/id/draft-ietf-itrace-04.txt.

[BEL89] S. M. Bellovin. “Security Problems in the TCP/IP Protocol Suite”. Computer Communications Review, Vol. 2, No. 19, pp. 32–48, Apr. 1989.

[BEZ11] B. Bruhadeshwar, S. S. Kulkarni, and A. X. Liu. “Symmetric Key Approaches to Securing BGP -- A Little Bit Trust is Enough”. *IEEE Transactions on Parallel and Distributed Systems*, Vol. 22, Issue 9, pp. 1536-1549, Sept. 2011.

[BLA06] D. Blazakis and J. S. Baras. “Analyzing BGP ASPATH behavior in the Internet”. *In Proceedings of the. 9th IEEE Global Internet Symposium*, pp. 468-473, Apr. 2006.

[BLU05] L. Blunk, J. Damas, F. Parent, and A. Robachevsky. “Routing Policy Specification Language Next Generation (RPSLng)”, RFC 4012, Internet Engineering Task Force, Mar. 2005.

[BRA98] K. A. Bradley, S. Cheung, N. Puketza, B. Mukherjee, and R. A. Olsson. “Detecting disruptive routers: a distributed network monitoring approach”. *IEEE Network*, Vol. 12, Issue 5, pp. 50–60, Sep/Oct. 1998.

[BRO08] M. A. Brown. “Pakistan hijacks YouTube”. *Renesys Blog*. Online, Feb. 2008. Available from http://www.renesys.com/blog/2008/02/pakistan-hijacks-youtube-1.shtml.

[BUS10] R. Bush. “The RPKI & Origin Validation”. *Presentation in RIPE 60 Meetings*. Online, May 2010. Available from http://www.ripe.net/ripe/meetings/ripe-60/presentations/Bush-The_RPKI_Origin_Validation.pdf.

[BUS11] R. Bush. “RPKI-Based Origin Validation Operation”. IETF Draft, Jul. 2011, http://tools.ietf.org/html/draft-ietf-sidr-origin-ops-10.

[BUT05] K. Butler, T. Farley, P. McDaniel, J. Rexford. “A Survey of BGP Security”. AT&T Tech Report. Online, Apr. 2005. Available from http://www.potaroo.net/papers/phd/refs/07-bgpsecuritysurvey.pdf.

[BUT06] K. Butler, P. McDaniel, and W. Aiello. “Optimizing BGP security by exploiting path stability”. *In Proceedings of the 13th ACM Conference Computer and Communications Security (CCS’06)*, pp. 298-310, Nov. 2006.

[BUT10] K. Butler, T. Farley, P. McDaniel, and J. Rexford. “A survey of BGP security issues and solutions". *Proceedings of the IEEE*, Vol. 98, Issue 1, pp. 100-122, Jan. 2010.

[CAE05] M. Caesar and J. Rexford. “BGP routing policies in ISP networks”. *IEEE Network*, Vol. 19, Issue 6, pp. 5-11, Nov. 2005.

[CHE97] S. Cheung. “An efficient message authentication scheme for link state routing”. *In Proceedings of the 13th Annual Computer Security Applications Conference (ACSAC 1997)*, pp. 90–98, Dec. 1997.

[CIT11] L. Cittadini, G. D. Battista, M. Rimondini, and S. Vissicchio. “Wheel + Ring = Reel: The Impact of Route Filtering on the Stability of Policy Routing”. *IEEE/ACM Transactions on Networking*, Vol. 19, Issue 4, pp 1085-1096, Aug. 2011.

[DIE06] T. Dierks and E. Rescorla. “The Transport Layer Security (TLS) Protocol Version 1.1”, RFC 4346, Internet Engineering Task Force, Apr. 2006.

[DUA08] Z. Duan, X. Yuan, and J. Chandrashekar. "Controlling IP Spoofing through Interdomain Packet Filters". *IEEE Transactions on Dependable and Secure Computing*, Vol. 5, No. 1, pp. 22-36, Jan. 2008.

[FIP02] U.S. National Bureau of Standards. “Secure Hash Standard”. FIPS PUB 180-2, Aug. 2002.

[GOL08] S. Goldberg, S. Halevi, A. D. Jaggard, V. Ramachandran, and R. N. Wright. “Rationality and traffic attraction: Incentives for honest path announcements in BGP". *In Proceedings of the ACM SIGCOMM 2008 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM 2008)*, pp. 267-278, Aug. 2008.

[GOL10] S. Goldberg, M. Schapira, P. Hummon and J. Rexford. “How secure are secure interdomain routing protocols”. *In Proceedings of the ACM SIGCOMM 2010 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM 2010)*, *pp. 87-98,* Aug. 2010.

[GOO03] G. Goodell, W. Aiello, T. Griffin, J. Ioannidis, P. McDaniel, and A. Rubin. “Working around BGP: An incremental approach to improving security and accuracy of interdomain routing”. *In Proceedings of the Network and Distributed System Security Symposium (ISOC NDSS’03)*, pp. 75–85, Feb. 2003.

[GRE02] B. Green. “BGP security update: Is the sky falling?’’. *In Presentation Archives of the North Americans Network Operators’ Group Meeting (NANOG25)*, June 2002.

[GRI05] T. Griffin and G. Huston. “BGP Wedgies,” RFC 4264 (Informational), Internet Engineering Task Force, Nov. 2005.

[HAR98] D. Harkins, D. Carrel, The Internet Key Exchange (IKE), RFC 2409, Internet Engineering Task Force, Nov. 1998.

[HAW96] J. Hawkinson and T. Bates. “Guidelines for Creation, Selection, and Registration of an Autonomous System (AS)”. RFC 1930, Internet Engineering Task Force, Mar. 1996.

[HEF98] A. Heffernan. “Protection of BGP Sessions via the TCP MD5 Signature Option”. RFC 2385, Internet Engineering Task Force, Aug. 1998.

[HEJ08] H. Li. “Method and System for Verifying Update Information in BGP”. US Patent 7,826,456, Nov. 2010.

[HU04] Y. C. Hu, A. Perrig, and M. Sirbu. “SPV: secure path vector routing for securing BGP”. *In Proceedings of the ACM SIGCOMM 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 179-192, Aug. 2004.

[HU06] Y. C. Hu, D. McGrew, A. Perrig, B. Weis, and D. Wendlandt. “(R)Evolutionary bootstrapping of a global PKI for securing BGP”. *In Proceedings of the 5th ACM Workshop on Hot Topics in Networks (Hotnets-V)*, Nov. 2006.

[HU07] X. Hu and Z. Mao. “Accurate Real-time Identification of IP Prefix Hijacking,” *IEEE Symposium on Security and Privacy*, pp. 3–17, May 2007.

[HUS09a] G. Huston. “BGP routing table resource pages”. Online, Jun. 2011. Available from http://bgp.potaroo.net/as2.0/bgp-active.html.

[HUS09b] G. Huston. “BGP in 2008”. *The ISP Column*. Online, Mar. 2009. Available from http://www.potaroo.net/ispcol/2009-03/bgp2008.html.

[HUS10] G. Huston, G. Michaelson. “Validation of Route Origination using the Resource Certificate PKI and ROAs”. IETF Draft, Nov. 2010, http://tools.ietf.org/html/draft-ietf-sidr-roa-validation-10.

[HUS11a] G. Huston, M. Rossi, G. Armitage. "Securing BGP - A Literature Survey", *IEEE Communications Surveys & Tutorials*, Vol. 13, Issue 2, pp. 199-222, 2011.

[HUS11b] G. Huston. “Addressing 2010”. *The ISP Column*. Online, Jan. 2011. Available from http://www.potaroo.net/ispcol/2011-01/addresses-2010.html.

[HUS11c] G. Huston. “The 32-bit AS Number Report”. Online, Jun. 2011. Available from http://www.potaroo.net/tools/asn32/.

[HUS11d] G. Huston, R. Bush. "Securing BGP with BGPsec". *The ISP Column*. Online, Jul. 2011. Available from http://www.potaroo.net/ispcol/2011-07/bgpsec.html.

[HUS99] G. Huston. “Interconnection, peering and settlements-part I”. *The Internet Protocol Journal*, Vol. 2, No. 1, Mar. 1999.

[ISO92] ISO. “Intermediate System to Intermediate System Inter-Domain Routing Information Exchange Protocol”. DIS 10747, Jul. 1992.

[ISR09a] J. Israr, M. Guennoun, and H. T. Mouftah. "Mitigating IP Spoofing by Validating BGP Routes Updates". *International Journal of Computer Science and Network Security*, Vol. 9, No. 5, pp. 71-76, May 2009.

[ISR09b] J. Israr, M. Guennoun, and H. T. Mouftah. "Credible BGP- Extensions to BGP for Secure Networking". *In the Proceedings of the Fourth IEEE International Conference on Systems and Networks Communications (ICSNC 2009)*, pp. 212-216, Sep. 2009.

[ISR10] J. Israr, M. Guennoun, and H. T. Mouftah. "Credible-BGP: A Hybrid Cryptosystem to Secure BGP". *In the Proceedings of the IEEE Global Communications Conference (GLOBECOM 2010)*, pp. 1-6, Dec. 2010.

[ISR11] J. Israr, M. Guennoun, and H. T. Mouftah. "Analysis of Impact of Trust on Secure Border Gateway Protocol". *In the Proceedings of the Sixteenth IEEE Symposium on Computers and Communications (ISCC’11),* pp. 1099-1104, June 2011.

[KAR06] J. Karlin, S. Forrest, and J. Rexford. “Pretty good BGP: Improving BGP by cautiously adopting routes”. *In Proceedings of the 14th International Conference on Network Protocols*, pp. 290-299, Nov. 2006.

[KAU05] C. Kaufman. “Internet Key Exchange (IKEv2) Protocol”. RFC 4306, Internet Engineering Task Force, Dec. 2005.

[KEN00a] S. Kent, C. Lynn and K. Seo. “Secure Border Gateway Protocol (S-BGP)”. *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 4, pp 582-592, Apr. 2000.

[KEN00b] S. Kent, C. Lynn, J. Mikkelson, and K. Seo. “Secure Border Gateway Protocol (S-BGP) real world performance and deployment issues”. *In Proceedings of the ISOC Symposium of Network and Distributed System Security (NDSS 2000)*, Feb. 2000.

[KEN03a] S. Kent. “Securing the Border Gateway Protocol: A status update”. *In the Proceedings of the Seventh IFIP TC-6 TC-11 Conference on Communications and Multimedia Security (CMS 2003)*, pp. 40-53, Oct. 2003.

[KEN03b] S. T. Kent. “Securing the Border Gateway Protocol“. *Internet Protocol Journal*, Vol. 6, No. 3, 2003.

[KEN05] S. Kent and K. Seo. “Security Architecture for the Internet Protocol”. RFC 4301, Internet Engineering Task Force, Dec. 2005.

[KRA97] H. Krawczyk, M. Bellare, and R. Canetti. “HMAC: Keyed-Hashing for Message Authentication”. RFC 2104, Internet Engineering Task Force, Apr. 1997.

[KRU03] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur. “Topology-based detection of anomalous BGP messages”. *In Proceedings of the Sixth International Symposium on Recent Advances in Intrusion Detection (RAID 2003)*, pp. 17-35, Sep. 2003.

[KUH09] R. Kuhn and S. Liu. “Practical Interdomain Routing Security”. IT Professional, Vol. 11, No. 6, pp. 54-56, Nov. 2009.

[KUM93] B. Kumar and J. Crowcroft. “Integrating security in inter-domain routing protocols”. ACM SIGCOMM Computer Communications Review, Vol. 23, No. 5, pp. 36–51, Oct. 1993.

[LAD06] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang. “PHAS: A Prefix Hijack Alert System”. *In Proceedings of the 15th USENIX Security Symposium*, Jul. 2006.

[LEP11a] M. Lepinski, S. Turner, “An Overview of BGPSEC”. IETF Draft, Mar. 2011, http://tools.ietf.org/html/draft-lepinski-bgpsec-overview-00.

[LEP11b] M. Lepinski, S. Kent, “An Infrastructure to Support Secure Internet Routing”. IETF Draft, May 2011, http://tools.ietf.org/html/draft-ietf-sidr-arch-13.

[Li11] Q. Li, M. Xu, J. Wu, X. Zhang, P. C. Lee, and K. Xu. “Enhancing the Trust of Internet Routing with Lightweight Route Attestation”. *In the Proceedings of ACM Symposium on Information, Computer and Communications Security (ASIACCS)*, pp. 92-101, Mar. 2011.

[LIA10] Y. Liao, L. Gao, R. Guerin, and Z.-L. Zhang. “Inter-domain routing under diverse commercial agreements". *IEEE/ACM Transactions on Networking*, Vol. 18, Issue 6, pp. 1829-1840, 2010.

[LYN03] C. Lynn, J. Mikkelson, K. Seo. “Secure BGP (S-BGP)”. IETF Draft, Jun. 2003, http://tools.ietf.org/id/draft-clynn-s-bgp-protocol-01.txt.

[MAH02] R. Mahajan, D. Wetherall, and T. Anderson. “Understanding BGP misconfiguration”. *In Proceedings of the ACM SIGCOMM 2002 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 3-16, Aug. 2002.

[MCA09] C. McArthur and M. Guirguis, "Stealthy IP Prefix Hijacking: Don't Bite Off More Than You Can Chew," *In Proceedings of IEEE GLOBECOM 2009*, pp. 2480–2485, Nov. 2009.

[MEY07] D. Meyer, L. Zhang and K. Fall. "Report from the IAB Workshop on Routing and Addressing", RFC 4984, Internet Engineering Task Force, Sep. 2007.

[MOO06] D. Moore, C. Shannon, D. Brown, G. Voelker, and S. Savage. "Inferring internet Denial-of-Service activity". ACM Transactions on Computer Systems, Vol. 24, No. 2, pp. 115-139, May 2006.

[MUR03] S. Murphy. "BGP Security Vulnerabilities Analysis". IETF Draft, Mar. 2003, draft-murphy-bgpvuln-02.txt.

[MUR09] S. Murphy and S. Weiler. "Progress Toward Securing the Routing Infrastructure". *In Proceedings of the 2009 Cybersecurity Applications & Technology Conference for Homeland Security (CATCH '09),* pp. 39-48, Mar. 2009.

[NAR02] R. Naraine, "Massive DDoS attack hit DNS root servers". *eSecurity Planet Magazine*. Online, Oct. 2002. http://www.esecurityplanet.com/trends/article.php/1486981/Massive-DDoS-Attack-Hit-DNS-Root-Servers.htm.

[NAR08] C. Narasimham, J. Pradhan. "Evaluation of Performance Characteristics of Cryptosystem Using Text Files". *Journal of Theoretical and Applied Information Technology*, Vol. 4, No. 1, pp. 56-60, Jan. 2008.

[NEU11] P. G. Neumann. "Risks to the public". *ACM SIGSOFT Software Engineering Notes*, Vol. 34, Issue 2, pp. 15-24, Mar. 2009.

[NG04] J. Ng. "Extensions to BGP to Support Secure Origin BGP (soBGP)". IETF Draft, Apr. 2004, http://tools.ietf.org/id/draft-ng-sobgp-bgp-extensions-02.txt.

[NIC02] D. Nicol, S. Smith, and M. Zhao. “Efficient security for BGP route announcements”. Dartmouth Computer Science Technical Report TR-2003-440, 2002.

[NIC04] D. M. Nicol, S. W. Smith, and M. Zhao. “Evaluation of efficient security for BGP route announcements using parallel simulation”. *Simulation Modelling Practice and Theory*, Vol. 12, Issue 3-4, pp. 187–216, Jul. 2004.

[NIC10] M. O. Nicholes and B. Mukherjee. “A Survey of Security Techniques for the Border Gateway Protocol (BGP)”. *IEEE Communications Surveys & Tutorials*, Vol. 11, No. 1, pp. 52-65, Mar. 2008.

[NOR04] O. Nordstro¨m and C. Dovrolis. “Beware of BGP attacks”. *ACM SIGCOMM Computer Communication Review*, Vol. 34, No. 2, pp. 1–8, Apr. 2004.

[OLI09] R. Oliveira, M. Lad, and L. Zhang. “Understanding the Challenges in Securing Internet Routing”. *In the proceedings of the Ninth Annual International Symposium on Applications and the Internet (SAINT 2009)*, pp. 145-148, Jul. 2009.

[OOR07] P. C. V. Oorschot, T. Wan, and E. Kranakis. “On interdomain routing security and pretty secure bgp (psbgp)”. *ACM Transactions on Information and System Security (TISSEC)*, Vol. 10, No. 3, Jul. 2007.

[ORT09] S. Ortiz. “Securing the Internet’s Routing Infrastructure”. *IEEE Computer Magazine*, Vol. 42, No. 4, pp. 21-23, Apr. 2009.

[PAR01] K. Park and H. Lee. “On the Effectiveness of Route-Based Packet Filtering for Distributed DoS Attack Prevention in Power-Law Internets”. *In Proceedings of the ACM SIGCOMM 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 15-26, Aug. 2001.

[PAX01] V. Paxson. “An analysis of using reflectors for distributed denial-of-service attacks”. *ACM Computer Communications Review (CCR)*, Vol. 31, No. 3, Jul. 2001.

[PER88] R. Perlman. “Network layer protocols with Byzantine robustness”. *Ph.D. Dissertation*, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, Oct. 1988.

[POE11] I. Poese, S. Uhlig, M. A. Kaafar, B. Donnet, and B. Gueye. “IP geolocation databases: unreliable?”. *ACM SIGCOMM Computer Communication Review*, Vol. 41 Issue 2, pp. 53-56, Apr. 2011.

[QIU06a] J. Qiu and L. Gao. “Hi-BGP: A lightweight hijack-proof inter-domain routing protocol”. *Technical Report*. Department of Electrical and Computer Engineering, University of Massachusetts, 2006.

[QIU06b] S. Y. Qiu, F. Monrose, A. Terzis, and P. D. McDaniel. “Efficient techniques for detecting false origin advertisements in inter-domain routing”. *In the Proceedings of the 2nd IEEE Workshop on Secure Network Protocols (NPSEC),* pp. 12 - 19, Nov. 2006.

[RAG07] B. Raghavan, S. Panjwani, and A. Mityagin. “Analysis of the SPV Secure Routing Protocol: Weaknesses and Lessons”. *ACM SIGCOMM Computer Communication Review*, Vol. 37, No. 2, Apr. 2007.

[RAM10] A. Ramaiah, R. Stewart, and M. Dalal. “Improving TCP's robustness to blind in-window attacks”. RFC 5961, Internet Engineering Task Force, Aug. 2010.

[REK95] Y. Rekhter and T. Li. “A border gateway protocol 4 (BGP-4)”. RFC 177, Internet Engineering Task Force, Mar. 1995.

[REK96] Y. Rekhter, T. Li, and S. Hares. “A border gateway protocol 4 (BGP-4)”. RFC 4271, Internet Engineering Task Force, Jan. 2006.

[REY06] P. Reynolds, O. Kennedy, E. G. Sirer, and F. B. Schneider. “Securing BGP using external security monitors”. *Technical Report*. Cornell University, 2006.

[RIC00] M. Richtel. “Yahoo attributes a lengthy service failure to an attack”. *The New York Times on the Web*, Feb. 2000.

[RIV78] R. Rivest, A. Shamir, and L.M. Adelman. “A method for obtaining digital signatures and public-key cryptosystems”. *Communications of the ACM*, Vol. 21, No. 2, pp. 120-126, Feb. 1978.

[RIV92] R. Rivest. “The MD5 Message-Digest Algorithm”. RFC 1321, Internet Engineering Task Force, Apr. 1992.

[RSA91] RSA Laboratories, “How fast is the RSA algorithm?”. *Public-Key Cryptography Standards (PKCS)*, 1991.

[SAV00] S. Savage, D. Wetherall, A. Karlin, and T. Anderson. “Practical network support for IP traceback”. *ACM SIGCOMM Computer Communication Review*, Vol. 30, No. 4, Oct. 2000.

[SHU12] C. A. Shue, A. J. Kalafut, M. Gupta. “Abnormally Malicious Autonomous Systems and Their Internet Connectivity”. *IEEE/ACM Transactions on Networking*, Vol. 20, Issue 1, pp. 220-230, Feb. 2012.

[SEO01] K. Seo, C. Lynn, and S. Kent. “Public-key infrastructure for the secure border gateway protocol (S-BGP)”. *In Proceedings of the IEEE DARPA Information Survivability Conference and Exposition II (DISCEX’01)*, pp. 239-253, Jun. 2001.

[SID10] IETF. “Secure Inter-Domain Routing (SIDR)”. Online, Sep. 2010. Available from http://datatracker.ietf.org/wg/sidr/.

[SMI96] B. Smith and J. Garcia-Luna-Aceves. “Securing the border gateway routing protocol” *In Proceedings of the Global Telecommunications Conference (GLOBECOM '96)*, pp. 81-85, Nov. 1996.

[SMI98] B. Smith and J. Garcia-Luna-Aceves. “Efficient security mechanisms for the border gateway routing protocol”. *Computer Communications*, Vol. 21, No. 3, pp. 203–210, Jun. 1998.

[SNO01] A. Snoeren, C. Partridge, L. Sanchez, C. Jones, F. Tchakountio, S. Kent, and W. Strayer. “Hash-based IP traceback”. *In Proceedings of the ACM SIGCOMM 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 3-14, Aug. 2001.

[STE03] J. Stewart. “DNS cache poisoning - the next generation”. *Technical Report*, LURHQ Threat Intelligence Group, Jan. 2003.

[STE08] J. Stewart, T. Bates, R. Chandra, and E. Chen. “Using a Dedicated AS for Sites Homed to a Single Provider”. RFC 2270, Internet Engineering Task Force, Jan. 1998.

[SUB04] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. Katz. “Listen and whisper: Security mechanisms for BGP”. *In Proceedings of the 1st Symposium on Networked Systems Design and Implementation (NSDI 2004)*, pp. 127-140, Mar. 2004.

[THA98] R. Thayer, N. Doraswamy, and R. Glenn. “IP Security Document Roadmap”. RFC 2411, Internet Engineering Task Force, Nov. 1998.

[UORV] University of Oregon Route Views Project. Online, Jun. 2011. Available from http://www.routeviews.org/.

[WAN03a] L. Wang, X. Zhao, D. Pei, R. Bush, D. Massey, and L. Zhang. “Protecting BGP routes to top-level DNS servers”. *In Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS 2003)*, pp. 851-860, May 2003.

[WAN03b] F. Wang and L. Gao. “On inferring and characterizing internet routing policies”. *In Proceedings of the 3rd ACM SIGCOMM Conference on Internet Measuremen*, pp. 15-26, Oct. 2003.

[WAN05] T. Wan, E. Kranakis, and P. C. van Oorschot. “Pretty secure BGP (psBGP)”. Technical Report, School of Computer Science, Carleton University, 2005.

[WAN10] H. Wang and W. Hao. “Detection of Invalid BGP Routes”. *In the Proceedings of the 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*, pp. 1-4, Sep. 2010.

[WEN06] D. Wendlandt and I. Avramopoulos. “Don’t secure routing protocols, secure data delivery”. *In Proceedings of the 5th ACM Workshop on Hot Topics in Networks (Hotnets-V)*, Nov. 2006.

[WHI03] R. White. “Securing BGP through Secure Origin BGP (soBGP)”. *Internet Protocol Journal*, Vol. 6, No. 3, pp. 15-22, Sep. 2003.

[WHI06] R. White. “Deployment Considerations for Secure Origin BGP (soBGP)”. IETF Draft, Jun. 2006, http://tools.ietf.org/id/draft-white-sobgp-architecture-02.txt.

[YIN10] H. Yin, B. Sheng, H. Wang, and J. Pan. “Keychain-Based Signatures for Securing BGP”. *IEEE Journal on Selected Areas in Communications*, Vol. 28, No. 8, pp. 1308-1318, Oct. 2010.

[ZHA01a] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. F. Wu, and L. Zhang. “Validation of the multiple origin ASes conflicts through BGP community attribute”. IETF Draft, Nov. 2001, http://tools.ietf.org/id/draft-zhao-idr-moas-validation-00.txt.

[ZHA01b] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. F. Wu, and L. Zhang. "An analysis of BGP multiple origin AS (MOAS) conflicts". *In the Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement (IMW '01)*, pp. 31-35, Nov. 2001.

[ZHA05a] M. Zhao, S. Smith, and D. Nicol. "Evaluating the performance impact of PKI on bgp security". *In the Proceedings of the 4th Annual PKI Research Workshop (PKI'05)*, Apr. 2005.

[ZHA05b] M. Zhao, S. W. Smith, and D. M. Nicol. "Aggregated path authentication for efficient BGP security". *In Proceedings of the 12th ACM conference on Computer and Communications Security,* Vol. 3, pp. 128-138, Nov. 2005.

[ZHA09] Y. Zhang, Z. Zhang, Z. Morley, Z. Mao, and Y. Hu. "HC-BGP: A Light-weight and Flexible Scheme for Securing Prefix Ownership". *In the Proceedings of the 2009 IEEE/IFIP International Conference on Dependable Systems & Networks (DSN '09)*, pp. 23-32, Jun. 2009.

[ZHA10] Z. Zhang, Y. Zhang, Y. C. Hu, Z. M. Mao, and R. Bush. "iSPY: Detecting IP Prefix Hijacking on My Own". *IEEE/ACM Transactions on Networking*, Vol. 18, No. 6, pp. 1815-1828, Dec. 2010.

Appendix A

Confidence Intervals

Simulated quantities such as blocking probability are measured by taking the mean of a succession of g runs, each of long enough time to ensure uncorrelated results. All runs are identical and independent from each other. The g independent results will be represented by B_1, B_2, B_3, · · · , B_{g-1}, B_g.

$$\text{The Mean } \bar{B} = \frac{1}{g}\sum_{i=1}^{g} B_i \qquad \text{(A.1)}$$

However, the mean of the independent simulation runs $\bar{B}$ provide us with a single numerical value for the estimate of the expected value $E|B| = \mu$. In order to know how good is the estimate provided by $\bar{B}$ for the simulation results, it is necessary to compute the variance of V_b^2.

$$V_b^2 = \frac{1}{g-1}\sum_{i}^{g}(B_i - \bar{B}) \qquad \text{(A.2)}$$

Small V_b^2 indicates that the results are tightly clustered around $\bar{B}$, and we can be confident that $\bar{B}$ is close to the $E|B|$. On the other hand, if V_b^2 is large, the results are widely dispersed about $\bar{B}$ and we can not be confident that $\bar{B}$ is close to $E|B|$. Instead of seeking a single value to estimate $E|B|$, we can specify the interval of values that is highly likely to contain the true value of the parameter. We begin by specifying some high probability, say $1-\alpha$. We then find the interval $[L(B), U(B)]$ such that the probability:

$$P[L(B) \leq \mu \leq U(B)] = 1-\alpha \tag{A.3}$$

This interval contains the true value of the parameter with probability $1-\alpha$. Such an interval is $1-\alpha$ × 100% confidence interval.

Using the standard deviation and the *t* distribution table, the lower and upper limits of the 95% confidence interval can be calculated as follows:

$$\text{Lower Limit}= \bar{B} - \frac{\sigma t_{\left[\frac{\alpha}{2}, g-1\right]}}{\sqrt{g}} \tag{A.4}$$

$$\text{Upper Limit} = \bar{B} + \frac{\sigma t_{\left[\frac{\alpha}{2}, g-1\right]}}{\sqrt{g}} \tag{A.5}$$

where:

$\propto = 0.05$

g = number of observations

$\bar{B}$ = sample average

σ = sample standard deviation

$=\sqrt{V_b^2}$

$=\sqrt{\frac{1}{g-1}\sum_i^g (B_i - \bar{B})}$

The confidence interval means that 95% of the simulation results fall within the interval.

Throughout this thesis, the confidence interval is computed based on five independent runs. From the table of the *t* distribution, the $t_{\left[\frac{\alpha}{2},4\right]}$is found to be 2.776. It was observed that more than 95% of the results were within the calculated confidence interval for each experiment.